Many-Valued Real Functions Computable by Finite Transducers using IFS-Representations

by

MICHAL KONEČNÝ

A thesis submitted to the
The University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

School of Computer Science
The University of Birmingham
June 13, 2000

獻給
我的妻子
順霞

Abstract

We try to classify many-valued functions of type $I_1 \times \cdots \times I_n \rightrightarrows I_0$ (for some compact intervals $I_i \subseteq \mathbb{R}$) that can be exactly realized by a finite transducer using some stream representation of the real numbers. We restrict our study to the so-called IFS-representations where numbers are represented by infinite compositions of contracting functions on a compact interval. IFS-representations include the radix (e.g. decimal, signed binary) representations and representations based on Möbius transformations (=LFT's).

We prove that a many-valued function which is computable by a finite transducer using IFS-representations with contractions that contract strongly enough, is locally 'distorted'-affine wherever it is locally single-valued and continuously differentiable. In the proof we use a technique which lets us treat any sufficiently strong contraction as an affine contraction. The theorem entails that a function computable by a finite transducer using affine (resp. LFT-) representations, is locally equal to some n-ary affine function (resp. LFT).

We also prove that many-valued n-ary piecewise affine functions with rational coefficients are computable by a finite transducer using the signed binary representation. We exhibit an example of a finitely computable function which is strictly increasing, has zero derivative on a dense set, and infinite derivative on another dense set.

Acknowledgements

I would like to thank my supervisor Achim Jung for motivating me to start and continue in this research, for asking inspiring questions and giving relevant comments and also for kindly pointing my efforts back to the right direction.

My research study has been kindly sponsored by the School of Computer Science and the Committee of Vice-Chancellors and Principles of the Universities of the United Kingdom (CVCP).

I would also like to thank Lord Jesus for giving meaning to this research as well as everything in my life, and my wife Sylvia for her support and patience during my writing-up.

Contents

List of Figures

Chapter 1

Introduction

1.1 Problem specification

In many computer applications, there is a need to represent and compute with real numbers. Nevertheless, real numbers, being infinite objects in their substance, generally cannot be stored in a computer memory nor supplied to a digital computer process as an argument in finite time.

The traditional solution to this problem is to approximate real numbers by a fairly large finite set of rational numbers, usually the floating point numbers. In many cases, during a computation, this approximation significantly deviates from what it represents due to accumulated round-off errors (cf. [Ménissier-Morain 1996] for a thorough investigation).

One solution to this problem, as studied by numerical analysts, is to develop algorithms that are stable—despite round-off errors, they calculate a result with a guaranteed degree of accuracy. Another solution is to abandon the floating point and compute without round-off errors by means of some representation of real numbers that allows arbitrary accuracy—so called *exact* real number computation. We restrict ourselves to this kind of computation in this thesis.

This study aims to contribute to answering the following general question:

> *Which real number functions can be computed very cheaply?*

By 'very cheaply', one can, for example, mean that the time consumed grows linearly with the accuracy of the result. Another interpretation of 'very cheaply' is that the computation needs only a fixed amount of memory independently from the accuracy of the result. The latter is stronger than the former because whenever an infinite computation uses a fixed amount of memory, the amount of information that it communicates with its environment is asymptotically linear in time. The opposite is not true because, for example, push-down automata communicate information linearly in time but can have infinitely many configurations. We restrict ourselves to the study of *finite state* computation in this thesis.

The study of computation within a low-level complexity class such as linear time or finite state, depends strongly on the chosen computational model and representation[1]. Therefore, in the next two sections, we will carefully consider what computational models and what representations are available for exact real number computation.

This dependency on representation adds an incentive to study finite computability in as wide as possible classes of representations and models in order to be able to *compare* different representations and models in terms of their capability to compute with finite memory.

1.2 Computation with infinite objects

Any computation with exact real numbers (or any other infinite objects) requires some way to communicate information about them between processes. For example, suppose that a process should be able to compute the value of a function $f\colon \mathbb{R} \to \mathbb{R}$ for any input real number $r \in \mathbb{R}$. This requires a way to specify both the number r and the result $f(r)$. It is not possible to represent all elements of a continuum sized set of real numbers as finite strings because there are only countably many of such strings. As we have already mentioned, to specify an arbitrary real number, a potentially infinite amount of information needs to be transferred in some form.

How can an infinite object be passed to or from a process? Like in classical computation with finite objects, one can use a *one-way* channel to pass the infinite value. The only difference is that the passing takes forever and the argument has to be divided into infinitely many finite portions. We shall call this approach to computation with infinite objects *stream-based*. The idea to use this model of computation for exact real number computation is due to Wiedmer [1980] and has been developed since by many researchers.

Apart from generalizing the one-way value passing methods of finite object computation, to infinite values, it is helpful to consider more general, *two-way* passing mechanisms. The process which should receive a value, asks queries about it. The process which knows the value, answers these queries. Therefore we call this approach *query-based*.

Let us return to the situation of a process computing a function f. If we use query-based value passing, the process that computes f answers queries about the result value $f(r)$. Whenever it needs some information about the argument r, it can issue a query to the argument provider from which it will obtain an answer. In the stream-based approach, the argument r will come as an infinite stream which the process computing f reads and, at the same time, produces another infinite stream representing the value $f(r)$.

Query-based computation has the advantage that the value receiver can influence the form in which it will get the information. Even more importantly, the value receiver can ask only for

[1] First order computability is independent of the choice of representation as long as a set of basic functions is computable [Brattka 1999]. The class of polynomial time computable real functions is the same for the most common representations [Ko 1991].

that part of the infinite information which it actually needs. Nevertheless, the former advantage does not apply to the passing of real numbers: Since any useful partial information about a real number is essentially finite, the efficiency gain of query-based real number passing would not be very significant. However, for computation with more complicated infinite entities, like geometrical objects, and mainly (first or higher order) real number functions, two-way value passing should be significantly more efficient.

An example of query-based real number computation is the one based on B-adic numbers developed and used by Böhm et al. [1986, 1990] and Ménissier-Morain [1996]. In their approach, (roughly speaking) the query is a natural number n specifying the desired precision and the answer is an approximation of the value of the form $[(k-1)/B^n, (k+1)/B^n]$ coded by the natural number k. To make best use of such a computation, the value receiver needs to calculate the desired precision of its argument before inquiring about its value. This is not always optimal because sometimes the precision with which we need to know a quantity varies depending on its value.

Another, similar, example is the oracle Turing machine used by Ko [1991] as a basis for his complexity theory of real functions. An oracle plays the role of a value provider to a Turing machine. The machine itself acts as a value provider by answering queries about the value that it should compute. The queries about a number are of the same form as in the previous example but using dyadic numbers, i.e. with $B = 2$. Ko also defines second order oracle machines that can enquire about an argument which is a continuous function. His machine provides two kinds of queries about a continuous function: firstly, asking for a dyadic approximation of its range within a given dyadic interval and secondly, asking for a dyadic approximation of the modulus of continuity within a given dyadic interval.

Despite the advantages of query-based computation, in this thesis, we restrict ourselves to computation of first order real functions and stream-based value passing. It might be a subject of further research to extend the material in this thesis to query-based value passing as well as some form of higher order real number computation.

An important feature of computation with infinite objects, whether stream- or query-based, is that one sometimes has to consider the case when the value provider does not provide the value exactly and, for some reason, either stops providing information on the stream or fails to respond to a query. Thus the value remains *under-specified*. In the context of *domain theory* [cf. Abramsky & Jung 1994], one would say that the value is *approximate* or *partial*. Nevertheless, in this thesis, we will restrict ourselves to *precise* computation in which this does not happen.

1.3 Representing real numbers by sequences

Any representation of the real numbers (or any other space) by sequences induces the notions of computable real numbers and functions from the notions of computable sequences and stream functions (see e.g. [Weihrauch 1987, Weihrauch 1997] for precise definitions). Together with computability, a representation also induces a topology on the represented space [cf. Kreitz & Weihrauch 1985]. An *admissible* representation is one which induces the natural topology and is maximal among such representations with respect to continuous reduction of representations [cf. Weihrauch 1997].

An important feature of admissible stream representations of real numbers is that they are *redundant*. Thus, every real number can be represented by several, typically infinitely many different sequences. The observation has been formalized and proved by Mazenc [1993] and recently in a different context by Brattka & Hertling [2000].

The most common stream representations of the real numbers are the decimal or binary representations that are part of the family of radix representations. These representations are not redundant (with the minor exception of decadic, respectively dyadic, numbers which have two different representations). Therefore they are not admissible which means not suitable for exact computation. This had been observed already by Turing in the correction [1937] to [Turing 1935]. The standard solution to this problem is to add negative digits to the representation. (It was first used for serial most-significant-digit-first natural number arithmetic by Avizienis [1961]). In this way, one gets admissible representations. In the binary case, the representation is called *redundant* or *signed binary*. This is the most commonly used admissible real number representation.

Another, and more sophisticated, way to represent the real numbers is by using continued fractions. This idea is due to Gosper [1972] and has been developed e.g., in [Peyton Jones 1984, Kornerup & Matula 1995]. An infinite continued fraction

$$a_0 + \frac{b_0}{a_1 + \frac{b_1}{\cdots}}$$

with non-negative coefficients can be viewed as the following infinite composition of endofunctions on $[0, \infty]$:

$$\left(x \mapsto a_0 + \frac{b_0}{x}\right) \circ \left(x \mapsto a_1 + \frac{b_1}{x}\right) \circ \cdots \tag{1.1}$$

which converges to the constant function with the value of the continued fraction.

The radix representations can be viewed in a similar fashion when we restrict ourselves to infinite sequences of digits of the form $0.d_1d_2d_3\ldots$ For the B-ary representation, such a sequence represents a number $a \in [0, 1]$ and can be viewed as the following infinite composition of endo-functions on $[0,1]$:

$$\left(x \mapsto \frac{x + \mathfrak{d}_1}{B}\right) \circ \left(x \mapsto \frac{x + \mathfrak{d}_2}{B}\right) \circ \cdots \tag{1.2}$$

converging to the constant function with value a. The same is true for the signed B-ary representation with the exception that the functions are of type $[-1,1] \to [-1,1]$.

Vuillemin [1990] noticed that both radix and continued fraction representations in the forms 1.2 and 1.1, respectively, are special cases of representations which take as a basis infinite compositions of linear fractional transformations (LFT's), also called Möbius transformations, which are functions of the form $x \mapsto \frac{ax+c}{bx+d}$. Several representations based on LFT's were proposed by Edalat & Potts [1997, 1997] and algorithms for them were developed in [Potts 1998, Heckmann 2000].

A further generalization of LFT-representations are representations that use arbitrary contracting maps on some *base interval* in place of LFT's. These are called '*iterated function system*' (*IFS*) representations because arbitrary compositions of a collection of contracting maps corresponds to the well-established notion of IFS from fractal theory [Hutchinson 1981, Edalat 1995]. Another important sub-class of IFS-representations are *affine representations*, i.e. those that use affine contractions only.

The author is not aware of any non-LFT IFS-representation which has been seriously considered for exact real number arithmetic. Nevertheless, IFS-representations are used in this thesis as an abstraction from LFT-representations, in order to obtain a result for LFT-representations easier.

1.4 Similar research

As far as the author is aware, all of the previous research in the area of finite state exact real number computability lies within the scope of stream-based computation with radix or LFT-representations. We will use this setting as well, with one extension that we will consider almost arbitrary IFS-representations. Thus we will consider finite state transducers that compute with streams of symbols, called digits, interpreted as contractions. The fact that within this limited scope, the class of finitely computable functions still depends on the representation, can be illustrated by the example of finitely computable numbers. These are exactly those whose representation is periodic. However, the set of numbers that have periodic representation is different in different representations.

Let us now review the previously published results in this area that are known to the author.

Raney [1973] proved that any LFT with rational coefficients is computable using the LFT-representation with two digits which are interpreted as the following contracting maps on

$[0, \infty]$: $x \mapsto \frac{x}{1+x}$ and $x \mapsto x + 1$. This result is particularly interesting because the representation which Raney uses is not redundant as the ranges of the two contractions ($[0, 1]$ and $[1, \infty]$, respectively) overlap only by one number. Due to that, the representation is not admissible and therefore also not computationally viable—for example, it is not possible to compute addition with this representation by any means because this function is not even continuous in the induced topology.

Heckmann [1998*a*] has characterized the unary and partially the binary LFT's which can be computed with the representation of Edalat & Potts [1997] with finite state transducers. (Binary LFT's are functions of the form $(x, y) \mapsto \frac{axy+cx+ey+g}{bxy+dx+fy+h}$.)

Independently to the author's research, a research group in the Ukraine have been working on a similar problem. Very recently, they have published results that overlap with some of the results in this thesis[2]. Although their version of these results are stronger in some aspects, they are weaker in other aspects and at the same time apply only to one particular representation. Let us review the main results of Lisovik et al, relevant in our context.

All of the papers of Lisovik et al. which we will mention, use the framework of R^m_n-transducers (and their close derivatives) defined by Lisovik in [1987, 1990]. A R^m_n-transducer, similarly like the transducer which we will define in Definition 3.3.1 on page 36, is a state transition diagram in which each transition has n input and m output channels. Contrary to our transducer, theirs is a deterministic one. On each output channel, these hypothetical machines use a redundant variant of the binary representation different from signed binary. On input channels they use a *non-redundant* version of this representation. Due to this and the determinism of the transducer, the real functions computed in this framework are single valued everywhere apart from dyadic points.

Their redundant version of the representation uses infinite sequences of digits $\{\bar{2}, \bar{1}, 0, 1, 2\}$ with binary dot ∇. Non-negative real numbers are represented only by the digits $\{0, 1, 2\}$ while non-positive ones only by the digits $\{0, \bar{1}, \bar{2}\}$. Thus the representation covers the whole of $\mathbb{R}$. The representation is redundant thanks to the digits $\bar{2}$ and 2. These "redundant" digits are not used on the input.

In [Lisovik & Shkaravskaya 1995], it is stated and in [Lisovik & Shkaravskaya 1998], fully proved that whenever a *push-down* R^1_1-transducer computes a function f *differentiable* on an interval (a, b), then f is affine on (a, b). In the latter paper, the same technique was used to prove an even stronger theorem stating that a function computed by a finite R^1_1-transducer cannot be properly convex on any interval (a, b).

Apart from these limiting results, it has been proved that there are finite transducers computing functions with various interesting properties. In [Lisovik 1990], there is a finite R^1_1-transducer computing a function which is nowhere differentiable. In [Lisovik & Shkaravskaya 1998], there is a finite R^2_1-transducer computing a Peano curve, i.e. a continuous surjection of an interval

[2]The author was not aware of the results until he received a review for his paper in the middle of May 2000.

to a square. Moreover, this Peano curve has the coefficient of overlapping 3 which is the least possible one. An inverse of this Peano curve is also finitely computable.

Shkaravskaya showed in [1998] that any affine function $f\colon I_1 \times \cdots \times I_n \to I_1 \times \ldots \times I_n$ with rational coefficients defined on a compact[3] hyper-rectangle with rational coordinates is computable by a finite R^n_n-transducer. In [Lisovik et al. 1999], it is shown that any unary piecewise affine function with rational coefficients is computable by a R^1_1-transducer.

1.5 Scope of the main result

Throughout this thesis, we will be studying exact real number computation by transducers using an IFS-representation of real numbers for each input and output channel. Most of this thesis will be devoted to results of the kind that they *limit* the set of real functions computable by finite transducers, i.e. statements of the form:

> *Any function from function set $\mathcal{F}$ with property ϕ computable by a finite transducer using representations from class E has property ψ.*

Let us now discuss what the symbols $\mathcal{F}$, E, ϕ and ψ will range over:

– The function set $\mathcal{F}$ will always be of the form $I_1 \times \cdots \times I_n \rightrightarrows I_0$ where $I_0, \ldots, I_n$ are closed real intervals or closed extended real intervals (see Definition 6.1.4 on page 100). The symbol '$\rightrightarrows$' constructs the type of a *many-valued* function which we will discuss below. We therefore consider *first-order many-valued real functions* (unary or of multiple arity) defined on compact (extended) rectangles.

  The reason why we work with many-valued functions instead of single-valued functions only, is that there are important problems in analysis and applied analysis (see e.g. [Aubin & Cellina 1984, Aubin & Frankowska 1990]) whose solution is computable as a many-valued function but not as a single-valued function. For example, there is no computable single-valued zero finding algorithm for polynomials, even for the case of the set of polynomials $x^3 - x + a$ for $a \in [-1, 1]$ as noted by Weihrauch [1995]. However, there is a computable many-valued zero finding algorithm for polynomials [cf. Ko 1991, Schönhage 1982].

  It has to be stressed that many-valued functions can be computed in two fundamentally different ways: *literally* and *indeterministically*. A machine computing a many-valued function literally returns the whole set of *all* values of the function at the argument. A machine which computes a many-valued function indeterministically, outputs the representation of an arbitrary *one* of the values at the argument.

[3]The original assumption is a bit weaker but not substantially.

The reason why we do not use literal computation of many-valued functions in this thesis, is that this would require to output a representation of a set of numbers rather than a representation of a single number. Therefore, with our setup with representations of single real numbers, this approach is not available. Therefore, from now on, whenever we say that a machine computes a many-valued function, we mean that it computes it indeterministically.

Indeterministic computation, unlike literal computation, assumes that the function has at least one value in each point. We will call a many-valued function with this property *total*. The symbol '$\rightrightarrows$' above actually refers only to total many-valued functions. We use another symbol '$\rightrightarrows$' to construct functional types which allow also *partial* many-valued functions.

Indeterministic computation with real numbers can be realized by either or both of two factors:

1. non-determinism in the transitions of the machine
2. redundancy in the representation of the argument.

There is a slight but important variation of indeterministic computation: A many-valued function is computed *completely* by a machine if it is computed by it and, for every argument-value pair of the function, there is a possible computation of the machine which realizes this pair. For any machine, there is exactly one function computed by it completely. Any other function whose graph contains this canonical function, is also said to be computed by the same machine. Therefore there are infinitely many functions computed by almost[4] any machine.

– The class E of representations will be either the affine representations, the LFT-representations or the IFS-representations *non-slow on the interior*. In order to explain this notion, we have to explain a bit more about what exactly we mean by IFS-representations. We will not use the standard definition of a contraction which is relative to a particular distance measure. We will define *d-contractions* as increasing, twice differentiable maps with exactly one fixpoint whose derivative is nowhere zero nor infinite. We will show that every d-contraction is a standard contraction relative to some metric topologically equivalent to the Euclidean one.

The derivative in the fixpoint of a d-contraction is a number in the interval $(0, 1]$ and will be called its *strength*. We will observe (in Proposition 5.1.5 on page 77) that any d-contraction with strength $w < 1$, when iterated, converges to the fixpoint exponentially with basis w while a d-contraction with $w = 1$ converges only sub-exponentially. We will call d-contractions with $w = 1$ slow and the others non-slow. That an IFS-representation

[4]There is a machine which ignores its input and non-deterministically produces a random number. Such a machine computes only one function whose graph is the whole product of the domain and the range of the function.

is 'non-slow on the interior' means that each finite composition of its d-contractions is either not slow or its fixpoint is an endpoint of the base interval. There is no result in this thesis for IFS-representations that use a slow d-contraction with a fixpoint in the interior of the represented interval.

Let us note that affine representations are special cases of LFT-representations and LFT-representations are special cases of IFS-representations non-slow on the interior (cf. Proposition 6.2.9 on page 104).

– The property ϕ will always be one of *continuous differentiability* on some *single-valued* piece of the graph of f, called *thread*. In the unary case, we consider also functions that can have infinite derivative in some points. Nevertheless, in the many-valued case, we exclude infinite partial derivatives because their treatment is technically difficult and not essential for the strength of the result.

  It has to be noted that in the case of affine representations, the technique of Lisovik & Shkaravskaya [1998] could probably be generalized to get analogous limiting results for unary functions with the weaker assumption of differentiability of a thread or even convexness/concaveness of a thread. Nevertheless, the author does not see how this technique could be generalized to functions of multiple arguments or IFS- or LFT-representations.

– The property ψ corresponds to the class of representations E as follows. For affine (respectively LFT-) representations, ψ says: 'the thread from ϕ is affine (respectively an LFT)'. For IFS-representations non-slow on the interior, the condition ψ is more complicated and could be intuitively described as follows: $\psi \sim$ 'the thread from ϕ is affine relative to certain metrics on the intervals in the domain and the range of $\mathcal{F}$'. (This metric can vary for different transducers even though they use the same representations. Nevertheless, the metric has to be in a class of metrics generated from the representations in a certain way.)

The main limitation theorem in this thesis extends the limitation theorem of Lisovik & Shkaravskaya [1998] in three important aspects:

1. It applies to n-ary real functions for any $n > 0$, instead of unary functions only.
2. It applies to many-valued functions and not only to single-valued ones.
3. It does not apply only to a particular representation but to a wide class of representations including the important class of LFT-representations.

1.6 Outline of the thesis

Chapter 2 '*Basics in indeterministic many-valued analysis*' is devoted to the definitions connected with many-valued functions. Notions of continuity and differentiability which are appropriate for indeterministic many-valued computation, are pointed out and studied to some extent.

Chapter 3 '*Exact Real Number Computation with IFS-Representations*' formally introduces d-contractions, IFS-representations and transducers that compute over them. The chapter is concluded with important propositions which make an equation between reachable states of a transducer and the functions which are computed from those states.

Chapter 4 '*Finite Automata and Affine Representations*' contains two sections. The first one progresses towards the main limitation theorem for affine representations via weaker versions for strongly connected unary transducers and for arbitrary unary transducers.

The second section of Chapter 4 gives two examples of functions that are computable by finite automata using the signed binary representation on $[-1, 1]$. The first one is a function which is strictly increasing and has derivative equal to 0 on a dense set E_1 and at the same time has derivative equal to $+\infty$ on another dense set E_2. This example enriches the collection of similar examples of Lisovik and Shkaravskaya.

The second example consists of piecewise affine functions of type $[-1, 1]^n \to [-1, 1]$ with rational coefficients. This result is a strengthening of the results of Shkaravskaya [1998] and Lisovik et al. [1999] about n-ary affine and unary piecewise affine functions respectively. The results in this thesis and those in the above papers have been developed totally independently from one another.

We also characterize the class of *many-valued piecewise affine* functions computable by finite transducers using the signed binary representation.

Chapter 5 '*Finite Automata and IFS-Representations*' extends the limitation results from previous chapter to IFS-representations non-slow on the interior. It is based on the simple idea that each non-slow d-contraction can be viewed as an affine map using a different metric.

Finally, Chapter 6 '*Finite Automata and LFT-Representations*' formally introduces LFT-representations and applies to them the limitation result about IFS-representations.

Let us point out the *novel contributions* of this thesis, according to author's knowledge:

– We note that indeterministically computable *real* functions are upper-semi-continuous but not necessarily lower-semi-continuous.

– There is a *new proof technique* of the unary limitation theorem for affine representations which leads to a slightly weaker result than the previous one of Lisovik & Shkaravskaya [1998]. However, unlike theirs, our technique can be readily applied also to transducers and functions of *multiple arguments* and non-affine representations.

– All *three limitation theorems* (for affine, LFT-, IFS-representations) for functions with *multiple arguments* are new.

– The technique of *translating a continuous contraction* to an affine contraction is probably also new. Certainly the idea of using this translation to extend results about affine representations to IFS-representations is new.

– Last, but not least, the characterization of *multi-dimensional piecewise affine functions* computable by finite transducers is stronger than previously published results [Shkaravskaya 1998, Lisovik et al. 1999] and the characterization of *many-valued piecewise affine functions* is completely new.

Let us note that some parts of the material in this thesis have been published by the author as technical reports. [Konečný 1998] is an early version of the unary single-valued affine case and also includes a direct proof of the LFT unary case of the limitation theorem. The technical report [Konečný 2000*b*] is an expanded version of the article [Konečný 2000*a*] which contains the single-valued version of the main limitation theorem for affine representations as well as the proof that all piecewise affine functions with rational coefficients are finitely computable using the signed binary representation.

The reader is assumed to have some knowledge of standard, very elementary notions from the area of real analysis [Rudin 1953], topology in the context of computation [Smyth 1992], graphs [Ore 1962, Diestel 1997] and finite automata [Trakhtenbrot & Barzdin 1973].

Chapter 2

Basics in Indeterministic Many-valued Analysis

This chapter reviews some of the fundamental notions in many-valued analysis based on the idea of indeterministic function evaluation. It is only a very small part of set-valued analysis (see e.g. [Aubin & Frankowska 1990]) which is relevant to us.

As we have discussed in the introduction, a many-valued function evaluation can be viewed as set-valued or indeterministic. In the former case, one always considers the whole set of values that a function has at some point as a single unit. In the latter case, we imagine that the function evaluates to exactly one element of the set of possible values at a point. In this point of view, only *total* functions can be considered.

This distinction plays a crucial role in the choice of definition of continuity below. It will also lead us to define derivative and differential as well as differentiability as local properties of an argument-and-value pair rather than of an argument only. Nevertheless, on the meta-level, we will equate many-valued functions to set-valued functions as usual.

The definition of derivative and differentiability in this chapter is not substantially used later in this thesis in the sense that the main theorems could be proved well without it. The material is included for two reasons: 1. It can help the reader to familiarize themselves with many-valued functions and gain more insight into the subject. 2. We will be able to obtain some partial results later in the thesis that involve these notions.

2.1 Many-valued functions

Although we will consider only total many-valued functions on the level of computation, on the meta-level, we need to consider the partial ones as well to be able to define function inversion.

Definition 2.1.1. A *partial many-valued function* $f\colon X \rightrightarrows Y$ for some sets X and Y, is a function of type $f\colon X \to \wp(Y)$. A *(total) many-valued function* $f\colon X \rightrightarrows Y$ is a partial many-valued function with $f(x) \neq \emptyset$ for all $x \in X$.

In this thesis, we will not distinguish an ordinary function $f\colon X \to Y$ from its many-valued counterpart $g\colon X \rightrightarrows Y$ defined by $g(x) = \{f(x)\}$ for all $x \in X$.

The *graph* of a partial many-valued function $f\colon X \rightrightarrows Y$ is the set

$$\mathbf{Gr}\,(f) = \left\{(x, y) \in X \times Y \;\middle|\; y \in f(x)\right\}.$$

For a set $U \subseteq X$ and a function $f\colon X \rightrightarrows Y$, define $f(U) := \bigcup_{x \in U} f(x)$.

The *inverse* of a partial many-valued function $f\colon X \rightrightarrows Y$ is the function $f^{-1}\colon Y \rightrightarrows X$ defined by

$$f^{-1}(y) = \{x \in X \,|\, y \in f(x)\}\,.$$

Let f_1, f_2 be many-valued functions of the following types: $f_1\colon X \rightrightarrows Y$, $f_2\colon Y \rightrightarrows Z$. The *composition* $f_2 \circ f_1$, of f_1 with f_2 (or f_2 after f_1), is the many-valued function of type $X \rightrightarrows Z$ defined by

$$(f_2 \circ f_1)(x) = f_2(f_1(x)) = \bigcup_{y \in f(x)} f(y) = \left\{z \in Z \;\middle|\; (\exists y \in f_1(x))(z \in f_2(y))\right\}$$

Notice that composition of total functions is a total function.

2.2 Threads

We are going to define a handle for generalizing results about real number single-valued functions to many-valued ones. The idea is that something which is true for a single-valued function, might be proved true for any single-valued sub-function of a many-valued function. Such a single-valued sub-function we will call a *slice* and its special case a *thread*. We will now define these notions precisely.

Let $\mathcal{I}$ denote the set of all closed non-singleton intervals in $\mathbb{R}$.

Definition 2.2.1 (Slice of a many-valued function).
Let $f\colon (X, \tau) \rightrightarrows I$ be a many-valued function between a topological space (X, τ) and an interval $I \in \mathcal{I}$. A continuous function $h\colon J \to I$ from some connected open subset $J \subseteq X$ such that for every $\vec{x} \in J$, it holds $h(\vec{x}) \in f(\vec{x})$, is called a *slice* of f.

A thread will be defined as a slice which is uniformly separated from the rest of the graph of the function. We will also define a zone of a thread to be some canonical open neighbourhood of the graph of the thread within which there are no other points of $\mathrm{Gr}\,(f)$ than those in $\mathrm{Gr}\,(h)$. The choice of a precise definition of a zone is not important. Nevertheless, it is important that every thread has a zone around it and that we will be able to refer to it.

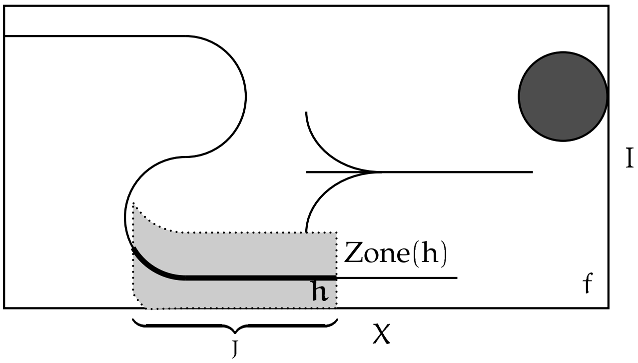

Figure 2.1: A thread and its zone.

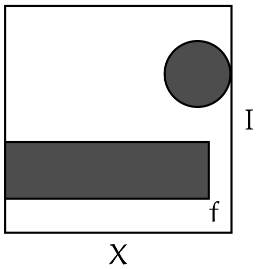

Figure 2.2: A function without a zone.

Definition 2.2.2 (Thread of a many-valued function).
A slice like in the context of previous definition, for which there is a number $\varepsilon > 0$ such that $(h(x) - \varepsilon, h(x) + \varepsilon) \cap f(x) = \{h(x)\}$ for all $x \in J$ is called a *thread* of f.

For each thread h, pick some canonical $\varepsilon_h > 0$ for which the statement above is true with $\varepsilon = \varepsilon_h$.

Define the *zone* of h to be the following open subset of $X \times I$ (see Figure 2.1):

$$\mathrm{Zone}(h) := \left\{(x, y) \;\middle|\; x \in J, y \in I \cap \big(h(x) - \varepsilon_h, h(x) + \varepsilon_h\big)\right\}.$$

Remark 2.2.3. In the context of the previous definition, $\mathrm{Zone}(h) \cap \mathrm{Gr}\,(f|_J) = \mathrm{Gr}\,(h)$.

Our results about many-valued functions will apply only to functions which have some threads. This does not mean that functions that have no thread are not interesting. Among these are, for example, the functions whose graph consists of a set of solids (cf. Figure 2.2) or whose graph is a Peano curve. The reason for this restriction is that our present techniques do not easily apply, up to the author's best knowledge, to functions that do not have threads.

2.3 Continuity

It turns out that the notion of *upper-semi-continuity* (see e.g. [Smyth 1992]) suits our computational model. Therefore we are defining continuity as upper-semi-continuity:

Definition 2.3.1 (Continuous many-valued function).
A many-valued function $f\colon (X, \tau_X) \rightrightarrows (Y, \tau_Y)$ between two topological spaces is *continuous* if for every $U \in \tau_Y$ the set $\{x \in X \,|\, f(x) \subseteq U\}$ is in $\{U \cap \mathrm{Dom}(f) \,|\, U \in \tau_X\}$.

Lower-semi-continuity which is defined by: $f^{-1}(U) \in \tau_X$ for each $U \in \tau_Y$, does not correspond to our model of computation because there are computable functions that are not lower-semi-continuous and lower-semi-continuous functions are rarely computable.

In fact, both of the continuities were considered already by Vietoris [1922]. In his terminology, upper-semi-continuity corresponds to the so-called 'box' or 'universal' inverse and lower-semi-continuity to the 'diamond' or 'existential' inverse of a relation:

$$\Box f(U) = f_{\forall}^{-1}(U) = \{x \in X \,|\, f(x) \subseteq U\}$$
$$\Diamond f(U) = f_{\exists}^{-1}(U) = \{x \in X \,|\, f(x) \cap U \neq \emptyset\}$$

According to Smyth [1992], the continuity defined by the existential inverse corresponds to partially correct non-deterministic computation while upper-semi-continuity to totally correct non-deterministic computation. In our model we are interested in totally correct computation.

As we will see later in Theorem 3.3.8 on page 42, upper-semi-continuity indeed corresponds to indeterministic definition of computation over the real numbers using IFS-representations. The proof of this theorem does not use any other properties of IFS-representations than the fact that they express a converging sequence of shrinking closed intervals and that the notions of closeness of sequences and of closeness of the represented numbers are tightly linked (see Lemma 3.2.14 on page 34). The conjecture is that something similar to Theorem 3.3.8 holds for transducers using any admissible stream-based representation of real numbers.

The author believes that with an additional property (which cannot be translated into an enhanced version of continuity), upper-semi-continuity exactly corresponds to expressibility by unlimited infinite transducers which use IFS-representations.

Let us now continue in our formal study of continuity. Like in the case of single-valued real functions, we will progress towards an alternative definition of continuity which is based on a local property:

Definition 2.3.2 (Continuity at a point).
A many-valued function $f\colon (X, \tau_X) \rightrightarrows (Y, \tau_Y)$ between two topological spaces is *continuous at a point* $x \in X$ if for every sequence $x_1, x_2, \ldots \in X$ converging to x and every sequence $y_1, y_2, \ldots \in Y$ with $y_j \in f(x_j)$ for all $j > 0$, converging to some y, it holds $y \in f(x)$.

A single-valued function is continuous if and only if its graph is closed provided that the codomain is Hausdorff. We will prove that the same holds also for (upper-semi-) continuity of many-valued functions. This observation is standard and appears e.g. in [Aubin & Cellina 1984].

Lemma 2.3.3. *The following are equivalent for a many-valued function whose codomain is a Hausdorff space:*

1. *it is continuous in all points of its domain,*
2. *its graph is closed,*
3. *it is continuous.*

For a subset U of a topological space, let U^o and $\overline{U}$ denote the interior and closure of U, respectively.

Proof. *1$\Longrightarrow$2:* Continuity of f: $(X, \tau_X) \rightrightarrows (Y, \tau_Y)$ in all points means that for every sequence $(x_1, y_1), (x_2, y_2), \ldots \in \mathrm{Gr}\,(f)$ such that $x_1, x_2, \ldots$ converge to x and $y_1, y_2, \ldots$ converge to y, also $(x, y) \in \mathrm{Gr}\,(f)$. By a standard topological observation, this is equivalent to $\mathrm{Gr}\,(f)$ being closed.

2$\Longrightarrow$3: Take any f: $(X, \tau_X) \rightrightarrows (Y, \tau_Y)$ whose graph $\mathrm{Gr}\,(f)$ is closed in $(X, \tau_X) \times (Y, \tau_Y)$, and take an open set $U \in \tau_Y$. Consider the closed set: $\mathrm{Gr}\,(f) \cap \big(X \times (Y \setminus U)\big)$. Its projection to X which is equal to $X \setminus \{x \in X \,|\, f(x) \subseteq U\}$, is also closed. Thus $\{x \in X \,|\, f(x) \subseteq U\}$ is open.

3$\Longrightarrow$1: Suppose that f: $(X, \tau_X) \rightrightarrows (Y, \tau_Y)$ is continuous and take an arbitrary $x \in X$, a sequence $x_1, x_2, \ldots \in X$ converging to x and a sequence $y_1, y_2, \ldots \in Y$ with $y_j \in f(x_j)$ for all j, converging to some y. For the sake of contradiction, assume $y \notin f(x)$.

Let $\Omega(y)$ be the set of all open neighbourhoods of y in (Y, τ_Y). For any $U \in \Omega(y)$, there is a number k_U, such that $y_i \in U$ for all $i > k_U$. Now $f(x_i) \not\subseteq (Y \setminus U)^o$ for all $i > k_U$. Thus all of x_i for $i > k_U$, are not in the set $\left\{z \in X \,\middle|\, f(z) \subseteq (Y \setminus U)^o\right\}$ which is open because f is continuous. Since all x_i belong to the complement of this open set, also x as their dense point, is in this complement. I.e. $x \notin \left\{z \in X \,\middle|\, f(z) \subseteq (Y \setminus U)^o\right\}$ and subsequently $f(x) \cap \overline{U}$ is not empty. Since this holds for any $U \in \Omega(y)$, it holds $f(x) \cap \bigcap_{U \in \Omega(y)} \overline{U} \neq \emptyset$, i.e. there is $y' \in f(x)$ which is in $\overline{U}$ for all $U \in \Omega(y)$.

We will prove $y' = y$ by contradiction. Assuming $y \neq y'$, we can find two disjoint open sets U, U' such that $y \in U$ and $y' \in U'$ because (Y, τ_Y) is Hausdorff. By the definition of y', we get $y' \in \overline{U}$ which is a contradiction with $U \cap U' = \emptyset$. ☐

2.4 Differentiability of unary real many-valued functions

The derivative of a real single-valued function expresses its 'speed of change' or 'slope' at a point. For a many-valued function, we could perhaps define the derivative at a point of the graph as all possible slopes of slices going through that point. In fact, we will do something very similar but slightly more general. We want to consider also infinite derivative like in the single-valued case. However, there is one case of infinite derivative which cannot be covered

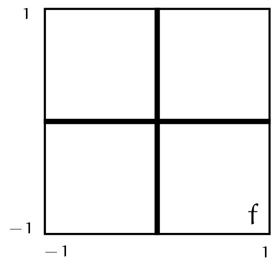

Figure 2.3: A function with derivative ∞ in points not on a slice.

by a slice. For example, we would like the function $x \mapsto \{0\}$ for $x \in [-1, 1] \setminus \{0\}$ and $0 \mapsto [-1, 1]$ (see Figure 2.3), to have a derivative equal 0 in all points $(x, 0)$ for $x \neq 0$, to ∞ in all points $(0, x)$ for $x \neq 0$, and to $\{0, \infty\}$ in the point $(0, 0)$. But the infinite derivative of this function at the points $(0, x)$ cannot be characterised by a slice because, locally, the graph of the function is a perpendicular line.

This also implies that it is not possible to distinguish $+\infty$ from $-\infty$ like in the single-valued case. It also makes sense not to distinguish the two infinite derivatives because they indicate the same slope. Let us now formalize the set of real numbers extended by a single ∞.

Definition 2.4.1 (Extended real numbers).
The one point compactification $\mathbb{R}^* = \mathbb{R} \cup \{\infty\}$ of $\mathbb{R}$ is called the *extended real numbers*. Addition, negation, multiplication and inversion of the real numbers are extended as many-valued functions to the new infinity value as follows:

$$\infty + x = \infty \text{ for all } x \in \mathbb{R}, \qquad \infty + \infty = \mathbb{R}^*$$
$$-\infty = \infty,$$
$$\infty \cdot x = \infty \text{ for all } x \in (\mathbb{R} \cup \{\infty\}) \setminus \{0\}, \qquad \infty \cdot 0 = \mathbb{R}^*$$
$$\infty^{-1} = 0 \qquad 0^{-1} = \infty$$

The values of $\infty + \infty$ and $0 \cdot \infty$ are usually undefined. We choose to define addition and multiplication many-valued at these points. One of the motivations for this is that such addition and multiplication are continuous functions of type $\mathbb{R}^* \times \mathbb{R}^* \rightrightarrows \mathbb{R}^*$.

Instead of slices, we will consider all *curves* going through a point of a many-valued function in order to define its derivative.

Definition 2.4.2 (Curve in a many-valued function).
A *curve in a many-valued function* $f\colon I_1 \rightrightarrows I_0$ on intervals $I_0, I_1 \in \mathcal{I}$, *going though the point* $(x, y) \in \mathrm{Gr}(f)$ is an injective continuous function $h\colon [-1, 1] \to I_1 \times I_0$ such that $h(0) = (x, y)$ and $h(t) \in \mathrm{Gr}(f)$ for every $t \in [-1, 1]$.

The curve h is called *normalized* if it is differentiable in 0 and the vector $(e, d) := h'(0)$ has the norm $\sqrt{e^2 + d^2}$ equal to 1.

Definition 2.4.3 (Derivative of a many-valued unary real function).
Let $f\colon I_1 \rightrightarrows I_0$ for some $I_1, I_0 \in \mathcal{I}$ and let $(x, y) \in \operatorname{Gr}(f)$. Denote by $f'_{(y)}(x)$ and call the *many-valued derivative* of f in (x, y), the following set

$$\left\{ \frac{e}{d} \;\middle|\; (d, e) = h'(0),\ h \text{ is a normalised curve in } f \text{ going through } (x, y) \right\}.$$

For any $x \in X$ and $U \subseteq f(x)$, let $f'_{(U)}(x) := \bigcup_{y \in U} f'_{(y)}(x)$. Similarly, for any $y \in Y$ and $U \subseteq X$ such that $y \in f(x)$ for all $x \in U$, define $f'_{(y)}(U) := \bigcup_{x \in U} f'_{(y)}(x)$.

In a naive description, the derivative consists of all possible values (in $\mathbb{R}^*$) of the slope of all curves h in f that go through the point (x, y).

Notice that unlike traditional single-valued derivative, the many-valued derivative is always defined, but the derivative can be anything between the empty set and the continuum-sized set $\mathbb{R}^*$.

Therefore the existence of a derivative cannot be taken as a definition of differentiability. We will use an analogy of the characterization of single-valued differentiability by the limit

$$\lim_{j \to \infty} \frac{f(x_j) - f(x)}{x_j - x}$$

for sequences $x_j \to x$ with $x_j \neq x$ for all j. In single-valued analysis, it holds that if all of these limits are defined and are equal to one and the same value, then f is differentiable in x and its derivative is that common value of the limits. In the many-valued case, the limits do not need to have a common value but each of the values will be required to agree with the derivative of some curve.

Definition 2.4.4 (The weak derivative).
For a many-valued function $f\colon I_1 \rightrightarrows I_0$, and a point $(x, y) \in \operatorname{Gr}(f)$, the set of all possible values of the following limit is denoted by $f^{wk}_{(y)}(x)$ and called the *weak derivative* of f in (x, y):

$$\lim_{j \to \infty} \frac{y_j - y}{x_j - x}$$

for all sequences $(x_1, y_1), (x_2, y_2), \ldots \in \operatorname{Gr}(f) \setminus \{(x, y)\}$ that converge to (x, y).

The following simple observation justifies the name 'weak derivative':

Lemma 2.4.5. *In the situation of the previous definition, it holds* $f'_{(y)}(x) \subseteq f^{wk}_{(y)}(x)$.

Proof. Choose any $t \in f'_{(y)}(x)$ and let h be a curve in f with $h(0) = (x, y)$, $h'(0) = (d, e)$ and $e/d = t$. Put $(x_j, y_j) := h(1/j)$ for all $j > 0$. Since h is continuous, it holds $\lim_{j\to\infty} x_j = x$ and $\lim_{j\to\infty} y_j = y$. Since h is injective, it holds $(x_j, y_j) \neq (x, y)$ for all j. Finally,

$$\lim_{j \to \infty} \frac{y_j - y}{x_j - x} = \frac{\lim_{j\to\infty} \frac{y_j - y}{1/j}}{\lim_{j\to\infty} \frac{x_j - x}{1/j}} = \frac{e}{d} = t$$

means that $t \in f^{wk}_{(y)}(x)$. $\square$

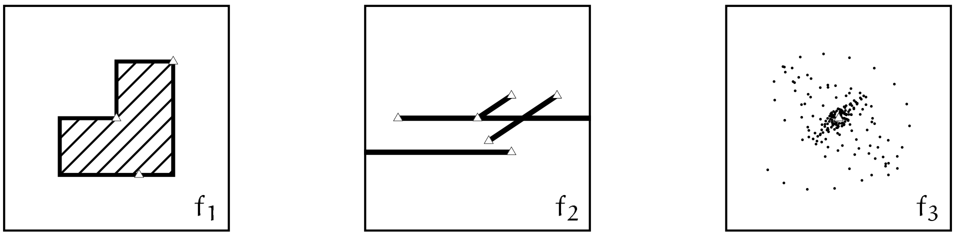

Figure 2.4: Many-valued functions differentiable and not differentiable at some points

The fact that weak derivative does not need to coincide with derivative is illustrated in Figure 2.4. The function f_1 has less values of derivative than values of weak derivative exactly in the points on the boundary of the solid. The functions f_2 and f_3 have this imbalance exactly in the points emphasized by a little white triangle. In f_2, each of the marked points has exactly one or two values of the weak derivative and one less value of the ordinary derivative. In the cloud in f_3, the isolated points have both weak and ordinary derivatives empty. The dense point of the cloud, provided that it belongs to the graph, can have a weak derivative equal to $\mathbb{R}^*$ and an empty ordinary derivative at the same time.

Definition 2.4.6 (Differentiability).
A many-valued function $f\colon I_1 \rightrightarrows I_0$ is called *differentiable in* $(x, y) \in \mathrm{Gr}(f)$ if $f'_{(y)}(x) = f^{\mathrm{wk}}_{(y)}(x)$.

It is differentiable in $x \in I_1$ if it is differentiable in (x, y) for all $y \in f(x)$.

The function f is called *differentiable* if it is differentiable on the whole of I_1, i.e. in all $x \in I_1$.

It is called *continuously differentiable* if it is differentiable and the many-valued function $x \mapsto f'_{(f(x))}(x)$ is continuous.

Remark 2.4.7 (Single-valued derivative).
For a single-valued function $f\colon I_1 \to I_0$, the ordinary derivative exists in $x \in I_1$ if and only if $f'_{(f(x))}(x)$ is a singleton set. In this case always $f'_{(f(x))}(x) = \{f'(x)\}$.

Let us conclude the section with a many-valued version of the rule for derivative of an inverse:

Lemma 2.4.8 (Derivative of inverse).
For the inverse f^{-1} *of any* $f\colon I_1 \rightrightarrows I_0$*, it holds:*

$$(f^{-1})'_{(x)}(y) = \frac{1}{f'_{(y)}(x)} \quad \textit{as well as} \quad (f^{-1})^{\mathrm{wk}}_{(x)}(y) = \frac{1}{f^{\mathrm{wk}}_{(y)}(x)} \tag{2.1}$$

for each $(x, y) \in \mathrm{Gr}(f)$.

Proof. For both equations, it is sufficient to prove the '$\supseteq$'-part of the set equality because the opposite follows from it by substituting f^{-1} in place of f.

In order to prove the first equation, take any curve h in f with $h(0) = (x, y)$ and $h'(0) = (d, e)$ such that e/d is defined and thus is in $f'_{(y)}(x)$. Observe that $\widehat{h}(y, x) := h(x, y)$ is a curve in f^{-1}

satisfying $\widehat{h}(0) = (y, x)$ and $\widehat{h}'(y, x) = (e, d)$. This means $d/e \in f'_{(x)}(y)$ and the first equation is proved.

Assume that $t \in f^{wk}_{(y)}(x)$, i.e. there is a sequence $(x_1, y_1), (x_2, y_2), \ldots \in \mathrm{Gr}(f) \setminus \{(x, y)\}$ converging to (x, y), such that the following limit converges to t:

$$t = \lim_{j \to \infty} \frac{y_j - y}{x_j - x}$$

Since it holds $x_j \in f^{-1}(y_j)$ for all $j > 0$, the following limit is in $(f^{-1})^{wk}_{(x)}(y)$:

$$\frac{1}{t} = \lim_{j \to \infty} \frac{x_j - x}{y_j - y}. \qquad \square$$

Corollary 2.4.9. *The inverse of a differentiable function is differentiable.*

2.5 Notations for many-valued functions of multiple arguments

In this thesis, we will often consider functions of the type $I_1 \times \cdots \times I_n \rightrightarrows I_0$ for $I_0, \ldots, I_n \in \mathcal{I}$. In order to reason about functions of more variables easier, let us explain some notation and conventions connected with functions of multiple arguments.

In the following, let all symbols I_i for $i \in \mathbb{N}$, denote intervals from $\mathcal{I}$.

Also, having fixed some $n \in \mathbb{N}$, let us denote the domain $I_1 \times \cdots \times I_n$ by the symbol V.

The symbol P will always denote a subset of the coordinates $\{1, \ldots, n\}$ and $\overline{P}$ will denote the complement of P: $\{1, \ldots, n\} \setminus P$.

For $\vec{x}, \vec{y} \in \mathbb{R}^n$, the term $|\vec{x} - \vec{y}|$ will denote the Euclidean distance of $\vec{x}$ and $\vec{y}$.

Definition 2.5.1 (Tuple projection).
For a point $\vec{x} = (x_1, \ldots, x_n) \in V$ and a set of coordinates P, let $\vec{x}_P$ denote the tuple $(x_i)_{i \in P}$.

Definition 2.5.2 (Function instantiation).
Let $f \colon V \rightrightarrows I_0$ be a function and P a set of coordinates. For any tuple of numbers $(x_i)_{i \in \overline{P}} \in \prod_{i \in \overline{P}} I_i$, denote

$$f_{P,(x_i)_{i \in \overline{P}}} \colon \prod_{i \in P} I_i \to I_0 : (x_i)_{i \in P} \mapsto f(x_1, \ldots, x_n),$$

the *instantiation* of f at coordinates P. If $P = \{k\}$ is a singleton set, the function can be also denoted by $f_{k,(x_i)_{i \neq k}}$.

If the value of the function f does not depend on the arguments in coordinates $\overline{P}$, we denote the instantiation of f at coordinates P simply by f_P.[1]

[1] By our conventions, for a tuple of functions $\vec{f} = (f_1, \ldots, f_m)$ each of which is of type $V \to I_0$, the notation $\vec{f}_P$ might mean two different things: the projection $(f_i)_{i \in P}$ as well as the pointwise application of function instantiation. For a tuple of functions we will always use the $\vec{f}_P$ notation to denote projection rather than instantiation.

2.6 Differential of many-valued functions

A hyper-curve of a function $f\colon V \rightrightarrows I_0$ could be defined analogously like the curve in the unary case which would lead to a proper treatment of infinite derivative. But this treatment is very complex because it would lead deeper to linear algebra over projective spaces. Moreover, as we mentioned in the introduction, considering infinite partial derivatives would not lead to any significantly stronger theorems in our case. Therefore, within this thesis, we restrict ourselves to differentiability of multi-dimensional real functions which excludes infinite partial derivatives. In this case, we can simply use slices instead of hyper-curves to define the differential.

Definition 2.6.1 (Differential).
Let $f\colon V \rightrightarrows I_0$ be a many-valued function and let $(\vec{x}, y) = (x_1, \dots, x_n, y) \in \mathrm{Gr}(f)$. Denote by $\mathbf{d}f_{(y)}(\vec{x})$ and call the *many-valued differential* of f in $(\vec{x}, y)$, the following set:

$$\left\{ \mathbf{d}h(\vec{x}) \;\middle|\; h\colon J \to I_0 \text{ is a slice of } f,\ \vec{x} \in J,\ h(\vec{x}) = y,\ h \text{ is differentiable in } \vec{x} \right\}$$

i.e. consisting of all linear forms $L\colon \mathbb{R}^n \to \mathbb{R}$ that are the differentials of a slice of f going through the point $(\vec{x}, y)$.

We proceed analogously to the unary case by defining a weak derivative based on sequences:

Definition 2.6.2 (Multi-dimensional weak derivative).
For $f\colon V \rightrightarrows I_0$ and $(\vec{x}, y) = (x_1, \dots, x_n, y) \in \mathrm{Gr}(f)$, define the *weak derivative* $\mathbf{d}f^{\mathrm{wk}}_{(y)}(\vec{x})$ of f to be the set of the possible values of the following limit:

$$\lim_{j \to \infty} \frac{(\vec{x}^{(j)} - \vec{x}, y^{(j)} - y)}{\left|\vec{x}^{(j)} - \vec{x}\right|} \tag{2.2}$$

for any sequence $(\vec{x}^{(1)}, y^{(1)}), (\vec{x}^{(2)}, y^{(2)}), \dots \in \mathrm{Gr}(f) \setminus (\vec{x} \times I_0)$ converging to $(\vec{x}, y)$.

Intuitively, the weak derivative consists of tuples $(\vec{e}, t)$ where $\vec{e} \in \mathbb{R}^n$ is a vector of length 1 representing an approaching direction to $\vec{x}$ and t is a possible slope of f when approaching y from the direction $\vec{e}$.

The following is a multi-dimensional analogy of Lemma 2.4.5.

Lemma 2.6.3. *For any continuous function $f\colon V \rightrightarrows I_0$ and any $(\vec{x}, y) \in \mathrm{Gr}(f)$ it holds: $(\vec{e}, L(\vec{e})) \in \mathbf{d}f^{\mathrm{wk}}_{(y)}(\vec{x})$ for every $L \in \mathbf{d}f_{(y)}(\vec{x})$ and every $\vec{e} \in \mathbb{R}^n$ with $|\vec{e}| = 1$.*

Proof. Let $h\colon J \to I_0$ be a slice with $\vec{x} \in J$, $h(\vec{x}) = y$ and $\mathbf{d}h(\vec{x}) = L$. Consider the points $\vec{z}^{(j)} := \vec{x} + \frac{1}{j}\vec{e}$ for all $j \in \mathbb{N}$. Since the sequence $\vec{z}^{(j)}$ converges to $\vec{x}$ and J is open, there is an index

$\widehat{\jmath}$ such that $\vec{z}^{(j)} \in J$ for all $j > \widehat{\jmath}$. Now the limit (2.2) can be applied to the sequence $\vec{x}^{(j)} := z^{(\widehat{\jmath}+j)}$, $y^{(j)} := h(\vec{x}^{(j)})$:

$$\lim_{j\to\infty} \frac{(\vec{x}^{(j)} - \vec{x}, y^{(j)} - y)}{\left|\vec{x}^{(j)} - \vec{x}\right|} = \lim_{j\to\infty} \frac{\left(\frac{1}{\widehat{\jmath}+j}\vec{e}, h(\vec{x} + \frac{1}{\widehat{\jmath}+j}\vec{e}) - h(\vec{x})\right)}{\left|\frac{1}{\widehat{\jmath}+j}\vec{e}\right|} =$$

$$\left(\frac{\vec{e}}{|\vec{e}|}, \lim_{j\to\infty} \frac{h(\vec{x} + \frac{1}{\widehat{\jmath}+j}\vec{e}) - h(\vec{x})}{\left|\frac{1}{\widehat{\jmath}+j}\vec{e}\right|}\right) = (\vec{e},\ \mathbf{d}h(\vec{x})(\vec{e})) = (\vec{e}, L(\vec{e})) \quad (2.3)$$

using the definition of the conventional differential $\mathbf{d}h(\vec{x})$. This means that $(\vec{e}, L(\vec{e})) \in \mathbf{d}f^{wk}_{(y)}(\vec{x})$, by Definition 2.6.2. $\square$

Definition 2.6.4 (Multi-dimensional differentiability).
A function $f\colon V \rightrightarrows I_0$ is called *differentiable in* $(\vec{x}, y) \in \mathrm{Gr}(f)$ if for any $(\vec{e}, t) \in \mathbf{d}f^{wk}_{(y)}(\vec{x})$ there is a $L \in \mathbf{d}f_{(y)}(\vec{x})$ such that $t = L(\vec{e})$.

The function f is differentiable in $\vec{x}$ if it is in $(\vec{x}, y)$ for all $y \in f(\vec{x})$. It is called *differentiable* if it is differentiable in every $\vec{x} \in V$. It is called *continuously differentiable* if it is differentiable and the partial derivative $\vec{x} \mapsto \bigcup_{y\in f(\vec{x})} \left(f_{k,(x_i)_{i\neq k}}\right)'_{(y)}(x_k)$ is continuous for every $k = 1, \dots, n$.

Remark 2.6.5. For a unary function $f\colon I_1 \rightrightarrows I_0$ it holds:

$$\mathbf{d}f_{(y)}(x) = \left\{x \mapsto t \cdot x \,\middle|\, t \in f'_{(y)}(x),\ t \neq \infty\right\}$$

$$\left\{t \,\middle|\, (1,t) \in \mathbf{d}f^{wk}_{(y)}(x) \text{ or } (-1,-t) \in \mathbf{d}f^{wk}_{(y)}(x)\right\} = f^{wk}_{(y)}(x) \setminus \{\infty\}$$

Therefore a unary function is differentiable as a multi-dimensional function if and only if it is differentiable as a unary function and its derivative does not include infinity at any point.

Lemma 2.6.6. *Let* $f\colon V \rightrightarrows I_0$ *be a function differentiable in* $(\vec{x}, y) \in \mathrm{Gr}(f)$*. The instantiation* $f_{P,\vec{x}_{\overline{P}}}$ *is a function differentiable in* $(\vec{x}_P, y)$*.*

Proof. Put $\ddot{f} := f_{P,\vec{x}_{\overline{P}}}$. Take some $(\vec{\ddot{e}}, t) \in \mathbf{d}\ddot{f}^{wk}_{(y)}(\vec{x}_P)$ and aim at proving that there is $\ddot{L} \in \mathbf{d}\ddot{f}_{(y)}(\vec{x}_P)$ with $\ddot{L}(\ddot{e}) = t$.

Take a sequence $(\vec{\ddot{x}}^{(j)}, y^{(j)}) \in \mathrm{Gr}\left(\ddot{f}\right)$ converging to $(\vec{x}_P, y)$ such that $\vec{\ddot{x}}^{(j)} \neq \vec{x}_P$ for all j, and

$$\lim_{j\to\infty} \frac{\left(\vec{\ddot{x}}^{(j)} - \vec{x}_P, y^{(j)} - y\right)}{\left|\vec{\ddot{x}}^{(j)} - \vec{x}_P\right|} = (\vec{\ddot{e}}, t).$$

For each j, denote by $\vec{x}^{(j)}$ the point in V whose coordinates are merged from $\vec{\ddot{x}}^{(j)}$ and $\vec{x}_{\overline{P}}$. Let $\vec{e} \in V$ be the extension of $\vec{\ddot{e}}$ with zeros in coordinates $\overline{P}$. Using the fact that $\left|\vec{x}^{(j)} - \vec{x}\right| = \left|\vec{\ddot{x}}^{(j)} - \vec{x}_P\right|$, we get

$$\lim_{j\to\infty} \frac{\left(\vec{x}^{(j)} - \vec{x}, y^{(j)} - y\right)}{\left|\vec{x}^{(j)} - x\right|} = (\vec{e}, t)$$

which means that $(\vec{e},t) \in \mathbf{d}f^{wk}_{(y)}(\vec{x})$. By differentiability of f in $(\vec{x},y)$, there is a slice $h\colon J \to I_0$ of f with $L(\vec{e}) = t$ for the differential $L := \mathbf{d}h(\vec{x})$.

Put $\ddot{L} := L_{P,(0)_{i\in\overline{P}}}$. Now it holds $\ddot{L}(\vec{\ddot{e}}) = L(\vec{e}) = t$. It remains to check only that $\ddot{L}$ is the differential of the thread $\ddot{h} := h_{P,\vec{x}_{\overline{P}}}$ in $(\vec{x}_P)$. This follows from elementary properties of the single-valued differential:

$$\left(\mathbf{d}\ddot{h}(\vec{x}_P)\right)((t_i)_{i\in P}) = \sum_{i\in P} \ddot{h}_i'(\vec{x}_P)\cdot t_i = \sum_{i\in P} h_i'(\vec{x})\cdot t_i = \left(\mathbf{d}h(\vec{x})\right)_{P,(0)_{i\in\overline{P}}}((t_i)_{i\in P}) = \ddot{L}((t_i)_{i\in P}) \quad \square$$

The following version of the chain rule is deliberately simplified by the strong assumption on functions g_i. Although a version without these assumptions can be proved, the proof would become too long and clumsy. This version is exactly what we will need in Chapter 5.

Lemma 2.6.7 (Differential of composition with multi-dimensional function).
Assume that

- $g_i\colon J_i \to I_i$ *is a continuous bijection between intervals* $J_i, I_i \in \mathcal{I}$, *differentiable in* $x_i \in J_i$ *for every* $i = 1,\dots,k$
- $g_0\colon I_0 \to J_0$ *is a continuous bijection between intervals* $I_0, J_0 \in \mathcal{I}$, *differentiable in* $y \in I_0$
- $f\colon I_1\times\cdots\times I_n \rightrightarrows I_0$ *is a continuous many-valued function differentiable in the point* $(\vec{g}(\vec{x}),y)$ *where* $\vec{g} = (g_1,\dots,g_n)$ *and* $\vec{x} = (x_1,\dots,x_n)$.

Then $g_0\circ f\circ\vec{g}$ *is differentiable in* $(\vec{x}, g_0(y))$ *and it holds*

$$\left(\mathbf{d}\left(g_0\circ f\circ\vec{g}\right)_{(g_0(y))}(\vec{x})\right)(\vec{t}) = g_0'(y)\cdot\left(\mathbf{d}f_{(y)}\left(\vec{g}(\vec{x})\right)\right)\left(g_1'(x_1)\cdot t_1,\dots,g_n'(x_n)\cdot t_n\right) \tag{2.4}$$

for any $\vec{t} = (t_1,\dots,t_n)\in\mathbb{R}^n$.

Proof. Take any $(\vec{e},t) \in \mathbf{d}(g_0\circ f\circ\vec{g})^{wk}_{(g_0(y))}(\vec{x})$ and aim at finding an $L \in \mathbf{d}(g_0\circ f\circ\vec{g})_{(g_0(y))}(\vec{x})$ with $L(\vec{e}) = t$ in order to prove the differentiability.

We will use the following method. Using the value $(\vec{e},t)$, we will derive a value of $\mathbf{d}f^{wk}_{(y)}(\vec{g}(\vec{x}))$, use the assumption that it is differentiable to obtain a linear form $K \in \mathbf{d}f_{(y)}(\vec{g}(\vec{x}))$ which will correspond to this value of the weak derivative. From K, we will be able to derive L which has the necessary properties.

First take some sequence $(\vec{x}^{(j)}, y^{(j)}) \in \mathrm{Gr}(f\circ\vec{g})\setminus(\vec{x}\times I_0)$ which converges to $(\vec{x},y)$ and satisfies

$$\lim_{j\to\infty}\frac{\left(\vec{x}^{(j)}-\vec{x},\, g_0(y^{(j)})-g_0(y)\right)}{\left|\vec{x}^{(j)}-\vec{x}\right|} = (\vec{e},t). \tag{2.5}$$

Now $(\vec{g}(\vec{x}^{(j)}), y^{(j)}) \in \mathrm{Gr}(f)\setminus(\vec{g}(\vec{x})\times I_0)$ is a sequence converging to $(\vec{g}(\vec{x}),y)$ by the continuity of g. Moreover, it holds

$$\lim_{j\to\infty}\frac{g_i(x_i^{(j)})-g_i(x_i)}{\left|\vec{x}^{(j)}-\vec{x}\right|} = \lim_{j\to\infty}\frac{g_i(x_i^{(j)})-g_i(x_i)}{x_i^{(j)}-x_i}\cdot\lim_{j\to\infty}\frac{x_i^{(j)}-x_i}{\left|\vec{x}^{(j)}-\vec{x}\right|} = g_i'(x_i)\cdot e_i$$

for every $i = 1, \dots, n$, which helps to derive

$$\lim_{j\to\infty} \frac{\left|\vec{g}(\vec{x}^{(j)}) - \vec{g}(\vec{x})\right|}{\left|\vec{x}^{(j)} - \vec{x}\right|} = \lim_{j\to\infty} \frac{\sqrt{\sum_{i=1}^{n} \left|g_i(x_i^{(j)}) - g_i(x_i)\right|^2}}{\left|\vec{x}^{(j)} - \vec{x}\right|} =$$

$$\sqrt{\sum_{i=1}^{n} \lim_{j\to\infty} \left(\frac{g_i(x_i^{(j)}) - g_i(x_i)}{\left|\vec{x}^{(j)} - \vec{x}\right|}\right)^2} = \sqrt{\sum_{i=1}^{n} (g_i'(x_i) \cdot e_i)^2} = \left|\left(g_1'(x_1) \cdot e_1, \dots, g_n'(x_n) \cdot e_n\right)\right|.$$

If $y^{(j)} \neq y$ for an infinite number of j, we can calculate

$$\lim_{j\to\infty} \frac{y^{(j)} - y}{\left|\vec{g}(\vec{x}^{(j)}) - \vec{g}(\vec{x})\right|} = \lim_{j\to\infty} \frac{g_0(y^{(j)}) - g_0(y)}{\left|\vec{x}^{(j)} - \vec{x}\right|} \cdot \frac{\left|\vec{x}^{(j)} - \vec{x}\right|}{\left|\vec{g}(\vec{x}^{(j)}) - \vec{g}(\vec{x})\right|} \cdot \frac{y^{(j)} - y}{g_0(y^{(j)}) - g_0(y)} =$$

$$\frac{t}{\left|\left(g_1'(x_1) \cdot e_1, \dots, g_n'(x_n) \cdot e_n\right)\right|} \cdot \frac{1}{g_0'(y)}$$

which holds also when $y^{(j)} = y$ for infinitely many j, because then $t = 0$ by (2.5) and both sides are equal to zero. Using all three equations above, we get a value of the weak derivative of f in $(\vec{g}(\vec{x}), y)$:

$$\lim_{j\to\infty} \frac{\left(\vec{g}(\vec{x}^{(j)}) - \vec{g}(\vec{x}), y^{(j)} - y\right)}{\left|\vec{g}(\vec{x}^{(j)}) - \vec{g}(\vec{x})\right|} = \frac{\left(\left(g_1'(x_1) \cdot e_1, \dots, g_n'(x_n) \cdot e_n\right), \frac{t}{g_0'(y)}\right)}{\left|\left(g_1'(x_1) \cdot e_1, \dots, g_n'(x_n) \cdot e_n\right)\right|}.$$

By the differentiability of f in $(\vec{g}(\vec{x}), y)$, we can find a slice $h\colon J \to I_0$ of f with $h(\vec{g}(\vec{x})) = y$, differentiable in $\vec{g}(\vec{x})$ such that the differential $K := \mathbf{d}h(\vec{g}(\vec{x}))$ fulfils:

$$K\left(g_1'(x_1) \cdot e_1, \dots, g_n'(x_n) \cdot e_n\right) = \frac{t}{g_0'(y)}.$$

Observe now that $g_0 \circ h \circ \vec{g}$ is a slice of $g_0 \circ f \circ \vec{g}$ going through $(\vec{x}, g_0(y))$ and differentiable in $\vec{x}$ by the conventional, single-valued chain rule. Using the chain rule for the partial derivatives of $g_0 \circ h \circ \vec{g}$, we finally get for $L := \mathbf{d}(g_0 \circ h \circ \vec{g})(\vec{x})$:

$$L(\vec{e}) = \sum_{i=1}^{n} (g_0 \circ h \circ \vec{g})_i'(\vec{x}) \cdot e_i = \sum_{i=1}^{n} g_0'(y) \cdot h_i'(\vec{g}(\vec{x})) \cdot g_i'(x_i) \cdot e_i =$$

$$g_0'(y) \cdot K\left(e_1 \cdot g_1'(x_1), \dots, e_n \cdot g_n'(x_n)\right) = t.$$

Having completed the proof of differentiability, let us prove equation (2.4). Consider the set H of all slices h of f that go through $(\vec{g}(\vec{x}), y)$ and are differentiable in $\vec{g}(\vec{x})$ as well as the set $\widehat{H}$ of all slices $\widehat{h}$ of $g_0 \circ f \circ \vec{g}$ that go through $(\vec{x}, g_0(y))$ and are differentiable in $\vec{x}$. The mappings $h \mapsto g_0 \circ h \circ \vec{g} : H \to \widehat{H}$ and $\widehat{h} \mapsto g_0^{-1} \circ \widehat{h} \circ \vec{g}^{-1} : \widehat{H} \to H$ are well-defined, injective and mutually inverse. Therefore they are bijections. Using these bijections and the equation

$$\left(\mathbf{d}(g_0 \circ h \circ \vec{g})(\vec{x})\right)(\vec{t}) = \sum_{i=1}^{n} (g_0 \circ h \circ \vec{g})_i'(\vec{x}) \cdot t_i = \sum_{i=1}^{n} g_0'(y) \cdot h_i'(\vec{g}(\vec{x})) \cdot g_i'(x_i) \cdot t_i =$$

$$g_0'(y) \cdot \sum_{i=1}^{n} h_i'(\vec{g}(\vec{x})) \cdot g_i'(x_i) \cdot t_i = g_0'(y) \cdot \left(\mathbf{d}h(\vec{g}(\vec{x}))\right)\left(g_1'(x_1) \cdot t_1, \dots, g_n'(x_n) \cdot t_n\right),$$

we can derive equation (2.4) straight from the definition of many-valued differential. $\square$

Chapter 3

Exact Real Number Computation with IFS-Representations

In this chapter, we will define a class of real number representations which generalizes the radix representations, continued fraction representations as well as all LFT-representations. The idea common to all these is that a sequence of intervals is represented by an infinite composition of contracting maps on the domain. The intervals in the sequence are then obtained as the ranges of the partial compositions of the infinite composition. IFS-representations will be defined along the same idea except that we will put fairly little requirements on the contracting maps.

The abbreviation IFS stands for 'iterated function system' as it is defined and used in fractal theory [Hutchinson 1981, Edalat 1995, Edalat 1996]. An IFS is a finite collection of continuous functions on some space. If all of the maps are contracting relative to some distance on the space, the IFS is hyperbolic. The largest set A which is equal to the union of the images $f(A)$ of all of the functions f in a hyperbolic IFS is called its attractor. We will not define IFS formally because we need only a modified special case of IFS. The reason for that is that we use IFS for a different purpose than fractal theory. The IFS which we need, operates on a real number interval and its attractor is the whole space.

3.1 D-contractions

The following definition of a contraction to be used in our version of IFS is specific to real numbers representation only. Unlike the traditional definition, it does not explicitly mention any distance measure on the domain. We call them <u>d</u>-contraction because we associate them with <u>d</u>igits of real number representations in this thesis.

Definition 3.1.1 (D-contraction).
An endo-function $f\colon I \to I$ on some interval $I \in \mathcal{I}$ is a *d-contraction* if and only if

1. f has a unique fixpoint
2. f is twice differentiable
3. for each $x \in I$ it holds $0 < f'(x) < \infty$

We will denote the unique fixpoint of f by $\mathbf{Fix_f}$. The derivative $f'(\mathrm{Fix}_f)$ at the fixpoint of a d-contraction f is called its *strength* and is denoted by $\mathbf{Strg_f}$. D-contractions with strength equal to 1 are called *slow*[1].

The maximal extension of a d-contraction f to intervals we will denote by $\overline{f}$:

$$\overline{f}\colon \mathcal{I}(I) \to \mathcal{I}(I) : [a, b] \mapsto \{f(x) \mid x \in [a, b]\}$$

where $\mathcal{I}(I)$ is the set of all closed subinterval of $I \in \mathcal{I}$ including the singletons.

The image $\overline{f}(I)$ of the base interval I is denoted by $[\![f]\!]$.

The set of all d-contractions on I is denoted by $\mathbf{dContr_I}$ and the set of all *affine* d-contractions on I by $\mathbf{dAff_I}$.

It is clear why a d-contraction is required to have exactly one fixpoint. It makes sense to require it to be monotonic because only this will guarantee that the extension to intervals will be strictly monotonic relative to interval inclusion. One might be wondering, though, what motivates condition 2 and why should the derivative of a d-contraction be not allowed to be equal to 0 or $+\infty$. The answer is hidden in Chapter 5 where these conditions will 'happen' to guarantee that a d-contraction can be viewed as an affine function in a nice way.

In the rest of this subsection, there are three lemmas exploring the very basic properties of d-contractions, in fact, showing that they behave like traditional contractions. The first two express the intuitive idea of contracting towards the fixpoint and the last lemma shows that finite iteration of a d-contraction is a d-contraction.

Lemma 3.1.2. *For any* $f \in \mathrm{dContr}_I$ *and any number* $x \in I$, *if* $x < \mathrm{Fix}_f$, *it holds* $x < f(x) < \mathrm{Fix}_f$ *and if* $\mathrm{Fix}_f < x$, *then* $\mathrm{Fix}_f < f(x) < x$. *Since* $\overline{f}([x, y]) = [f(x), f(y)]$ *for all* $x, y \in I$, *the extension* $\overline{f}\colon \mathcal{I}(I) \to \mathcal{I}(I)$ *has a unique fixpoint, namely* $[\mathrm{Fix}_f, \mathrm{Fix}_f]$.

Proof. Denote the endpoints of I by a, b: $I = [a, b]$. The function $z \mapsto f(z) - z$ has a unique root Fix_f, it is non-negative in a and non-positive in b. Assume $x < \mathrm{Fix}_f$. Then $a \leqslant x < \mathrm{Fix}_f$ and by the continuity of f, we get that the function $z \mapsto f(z) - z$ is positive on the whole of $[a, \mathrm{Fix}_f)$, and also in x. That means $x < f(x)$. The inequality $f(x) < \mathrm{Fix}_f$ follows from $x < \mathrm{Fix}_f$ and the fact that f is increasing. The proof of the dual implication is analogous. The rest of the lemma is obvious. □

[1]The closer its strength is to zero, the stronger the contraction contracts if iterated. See Corollary 3.1.7 below and Proposition 5.1.5 on page 77 for a justification of the name.

Lemma 3.1.3. *For any* $f \in \mathrm{dContr}_I$ *and any number* $x \in I$, *it holds*

$$\lim_{k\to\infty} f^k(x) = \mathrm{Fix}_f.$$

Proof. Assume $x \neq \mathrm{Fix}_f$ since the case $x = \mathrm{Fix}_f$ is trivial. By Lemma 3.1.2 the sequence $f^k(x)$ is strictly monotonic. Since it is also bounded, it has a limit. Denote it L. By continuity of f we get:

$$f(L) = f(\lim_{k\to\infty} f^k(x)) = \lim_{k\to\infty} f^{k+1}(x) = L$$

which implies $L = \mathrm{Fix}_f$. □

Lemma 3.1.4. *For any* $f \in \mathrm{dContr}_I$ *and any* $k \in \mathbb{N}$, $k > 0$, *the composition* f^k *is a d-contraction with* $\mathrm{Fix}_{f^k} = \mathrm{Fix}_f$.

Proof. From elementary analysis it follows that the composition f^k is twice differentiable with positive first derivative. All that needs to be shown is that Fix_f is its unique fixpoint. Firstly, it is a fixpoint: $f^k(\mathrm{Fix}_f) = f^{k-1}(f(\mathrm{Fix}_f)) = f^{k-1}(\mathrm{Fix}_f) = \cdots = \mathrm{Fix}_f$.

Secondly, suppose that $f^k(x) = x$ for some $x \in I$ and $x \neq \mathrm{Fix}_f$. W.l.o.g. assume $x < \mathrm{Fix}_f$. By Lemma 3.1.2, we get $x < f(x) < f^2(x) < \cdots < f^k(x) < \mathrm{Fix}_f$ which contains the contradiction $x < f^k(x)$. This proves the uniqueness of the fixpoint Fix_f. □

3.1.1 Strength of d-contractions

We will show that the number called strength is always in the range $(0, 1]$ and that it really can be viewed as some measure of contractivity of an iterated d-contraction. At this moment we do not have the tools to prove this in a very general case. Therefore we will prove it only for the affine case and generalize it fully in Chapter 5 as Proposition 5.1.5 on page 77.

Lemma 3.1.5. *The strength of a d-contraction* f *fulfils the inequality* $0 < \mathrm{Strg}_f \leqslant 1$.

Proof. By point (3) in the definition of d-contraction, we get $0 < \mathrm{Strg}_f < \infty$. For the sake of contradiction, suppose that $1 < \mathrm{Strg}_f < \infty$. Denote $I = [a, b]$. It holds either $a \neq \mathrm{Fix}_f$ or $b \neq \mathrm{Fix}_f$. Suppose first $a \neq \mathrm{Fix}_f$. By the definition of derivative, there is a point $c < \mathrm{Fix}_f$ near the fixpoint such that $(\mathrm{Fix}_f - f(c))/(\mathrm{Fix}_f - c) > 1$. This implies $f(c) < c$ which contradicts Lemma 3.1.2. In case $\mathrm{Fix}_f \neq b$, we can use the dual argument to achieve a contradiction. □

Although the following lemma is almost trivial, it is nice to have it pointed out because of its extensive use in the proofs in the following chapter.

Lemma 3.1.6. *Let* $f \in \mathrm{dAff}_I$ *and* $x, y \in I$. *For all* $k \in \mathbb{N}$ *it holds* $f^k(x) - f^k(y) = (x - y) \cdot \mathrm{Strg}_f^k$.

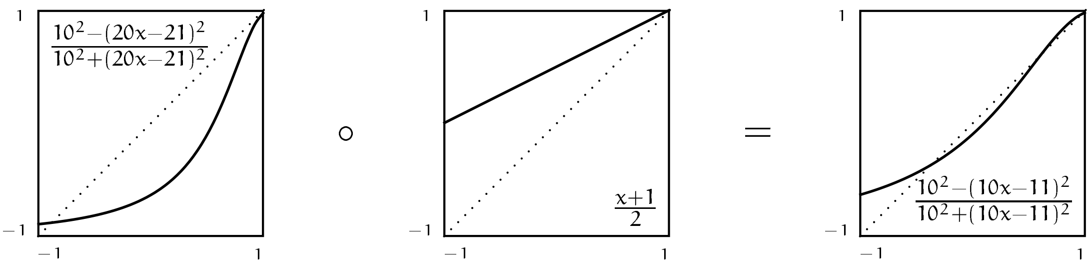

Figure 3.1: Graphs of two d-contractions and their composition

Proof. Denote $f(x) = w \cdot x + c$ where $w, c \in \mathbb{R}$. It holds $\mathrm{Strg}_f = w$ and $f(x) - f(y) = w(x - y)$ which yields the lemma when iterated. $\square$

Corollary 3.1.7 (Iterated affine d-contraction shrinks exponentially).
For $f \in \mathrm{dAff}_I$ *it holds* $\left|[\![f^k]\!]\right| = |I| \cdot \mathrm{Strg}_f^k$ *and* $\mathrm{Strg}_f < 1$ *(i.e.* f *is not slow).*

Proof. Denote $I = [a, b]$. The equation is a special case of Lemma 3.1.6 with $x = a$ and $y = b$. If $\mathrm{Strg}_f = 1$, then I would be a fixpoint of $\overline{f}$ which contradicts Lemma 3.1.2. $\square$

3.1.2 D-contractions vs. contractions

The notion of d-contraction differs from the standard notion of contraction on a real interval even if it fulfils points (2) and (3) in Definition 3.1.1. A contraction is defined as an endofunction which strictly decreases a distance measure of any two distinct points in the domain of the function. This means that a d-contraction does not need to be a contraction relative to the standard metric distance on $\mathbb{R}$ (see the leftmost function in Figure 3.1, for example). The fact that there is no single distance measure with respect to which all d-contractions would be contractions is demonstrated by the failure of d-contractions to be closed on composition:

Lemma 3.1.8 (Composition of d-contractions).
The functions

$$f_q\colon [-1, 1] \to [-1, 1] : x \mapsto \frac{10^2 - (20x - 21)^2}{10^2 + (20x - 21)^2}$$

$$\textit{and} \quad f_a\colon [-1, 1] \to [-1, 1] : x \mapsto \frac{x + 1}{2}$$

are d-contractions. But their composition

$$f_c = f_q \circ f_a : x \mapsto \frac{10^2 - (10x - 11)^2}{10^2 + (10x - 11)^2}$$

is not a d-contraction (cf. Figure 3.1 for an illustration).

Proof. Both functions f_q and f_a are twice differentiable. It is easy to see that f_a has a positive derivative and that it has a unique fixpoint equal to 1. Thus f_a is a d-contraction. The derivative of f_q is

$$f_q'(x) = \frac{-40(20x-21)\big(10^2+(20x-21)^2\big) - 40(20x-21)\big(10^2-(20x-21)^2\big)}{\big(10^2+(20x-21)^2\big)^2} = \frac{8000(21-20x)}{\big(10^2+(20x-21)^2\big)^2}$$

which is positive for $x \in [-1,1]$. The fixpoints of f_q are the roots of the cubic polynomial $P(x) = 400x^3 - 20x^2 + 121x + 341$. Its values in $-1, 0, 1$ are $-200, 341, 842$, respectively. This means that there is a root in $[-1, 0]$. The derivative $P'(x) = 1200x^2 - 40x + 121$ is always positive and therefore P has only one root and f_q has only one fixpoint.

The concatenation f_c has more than one fixpoint in $[-1,1]$ because it is continuous and it alternates along the identity diagonal:

$$f_c(-1) = -\tfrac{341}{541} > -1 \qquad f_c(0) = -\tfrac{21}{221} < 0 \qquad f_c(\tfrac{3}{4}) = \tfrac{1404}{1796} > \tfrac{1400}{1800} = \tfrac{7}{9} > \tfrac{3}{4}$$

□

Despite the difference between contractions and d-contractions, one can show that for every non-slow d-contraction on I, there is a metric on I equivalent to the standard one relative to which it is a contraction:

Proposition 3.1.9 (Non-slow d-contraction is a contraction).
Let $I \in \mathcal{I}$ *and* $f \in \mathrm{dContr}_I$, f *is not slow. There exists a function* $\mu_f \colon I \times I \to [0, \infty)$ *which is a metric on* I, *is equivalent to the standard metric* $(x, y) \mapsto |x - y|$, *and* f *is a contraction with respect to it.*

We will not be able to prove this until later as a corollary of Theorem 5.1.2 on page 76.

3.2 IFS-representations

We will define a version of IFS which will later turn into an IFS-representation when some additional conditions will be imposed upon it. For this reason, the definition below has a syntactic element, i.e. a set of digits, and a semantical element, i.e. a function which maps the digits to some d-contractions.

Definition 3.2.1 (D-IFS).
A tuple (I, Σ, D) where

1. $I \in \mathcal{I}$ is a *base interval*

2. Σ is a finite set of symbols called *digits*

3. $D\colon \Sigma \to \mathrm{dContr}_I$ is the interpretation of the digits

is called a *d-Iterated Function System* (d-IFS) over the interval I.

A d-IFS is *affine* if $D(\mathfrak{d})$ is affine for every $\mathfrak{d} \in \Sigma$.

The semantics of digits as d-contractions can be used to give meaning to sequences of digits via composition of their d-contractions as follows:

Definition 3.2.2 (Semantics of d-IFS).
For a d-IFS $\mathcal{S} = (I, \Sigma, D)$ we can define the extension $[\![\cdot]\!]_{\mathcal{S}} \colon \Sigma^{\infty} \to \mathcal{I}[I]$ of $\mathfrak{d} \mapsto [\![D(\mathfrak{d})]\!]$ for all arbitrary sequences of digits:

$$\begin{aligned}
[\![\epsilon]\!]_{\mathcal{S}} &= I \\
[\![\mathfrak{d}_1 \cdots \mathfrak{d}_n]\!]_{\mathcal{S}} &= \left(\overline{D(\mathfrak{d}_1)} \circ \cdots \circ \overline{D(\mathfrak{d}_n)}\right)(I) \\
[\![\mathfrak{d}_1\mathfrak{d}_2\mathfrak{d}_3 \cdots]\!]_{\mathcal{S}} &= \bigcap_{i \in \mathbb{N}} [\![\mathfrak{d}_1 \cdots \mathfrak{d}_i]\!]_{\mathcal{S}}
\end{aligned}$$

If there is no risk of confusion, we leave out the subscript of the semantic brackets $[\![]\!]$.

Remark 3.2.3. For a sequence of digits $\mathfrak{d}_1 \cdots \mathfrak{d}_n$ of a d-IFS (I, Σ, D) and any $k \in \mathbb{N}$, $1 \leqslant k \leqslant n$, it holds $[\![\mathfrak{d}_1 \cdots \mathfrak{d}_n]\!] \subseteq [\![\mathfrak{d}_1 \cdots \mathfrak{d}_k]\!]$

The semantic function $[\![\cdot]\!]$, in the finite case, returns the range of the composition of digit functions which is an interval. This interval is thought of as the information captured by the sequence of digits. This information is of the form: *'the number which is being represented, is within this interval'*. The previous definition expresses the idea that further digits in a sequence *refine* the information given by the preceding ones. Each successive digit restricts the domain of the function given by its preceding digits. Thus it also restricts the range of the new composite function to a smaller range (see Figure 3.3 on page 36 for an illustration).

The following definition is not a new one. It exactly corresponds to the weakly hyperbolic IFS from [Edalat 1996] whose attractor is the whole space, in this case a compact real interval. Also in the thesis [Potts 1998], there has appeared a definition of representations similar but more general than the following one although not explicitly called 'IFS' representations.

Definition 3.2.4 (IFS-representation).
A d-IFS $\mathcal{S} = (I, \Sigma, D)$ for which it holds:

1. $(\forall \alpha \in \Sigma^{\omega})(\exists r \in I)([\![\alpha]\!] = \{r\})$ (digits contract strongly enough)

2. $(\forall r \in I)(\exists \alpha \in \Sigma^{\omega})([\![\alpha]\!] = \{r\})$ (every number can be represented)

is called an *IFS-representation*[2] of I.

An IFS-representation in which $D(\mathfrak{d})$ is affine for all $\mathfrak{d} \in \Sigma$ is called *affine*.

The members I, Σ and D of an IFS-representation $\mathcal{S}$ can be referred to by $I_{\mathcal{S}}$, $\Sigma_{\mathcal{S}}$ and $D_{\mathcal{S}}$, respectively.

It is sufficient to demand that each number is covered by some digit instead of demanding condition 2:

Lemma 3.2.5 (Alternative definition).
The previous definition is equivalent to its variant in which condition 2 is replaced by the following:

2′. $(\forall r \in I)(\exists \mathfrak{d} \in \Sigma)(r \in [\![\mathfrak{d}]\!])$

Proof. Let (I, Σ, D) be an IFS-representation and $r \in I$. By condition 1, there is $\mathfrak{d}_1\mathfrak{d}_2 \cdots \in \Sigma^\omega$ with $[\![\mathfrak{d}_1\mathfrak{d}_2 \cdots]\!] = \{r\}$ which implies $r \in \overline{D(\mathfrak{d}_1)}(I) = [\![\mathfrak{d}_1]\!]$. Thus condition 2′ holds.

On the other hand, suppose that (I, Σ, D) is a d-IFS for which conditions 1 and 2′ hold and pick $r \in I$. Define two infinite sequences $r_1, r_2, \ldots \in I$ and $\mathfrak{d}_1, \mathfrak{d}_2, \ldots \in \Sigma$ inductively as follows: $r_1 := r$, and for each $i > 0$, let $\mathfrak{d}_i$ be a digit with $r_i \in [\![\mathfrak{d}_i]\!]$ (existence of which is guaranteed by condition 2′) and r_{i+1} be the number for which $D(\mathfrak{d}_i)(r_{i+1}) = r_i$. Now it holds $r \in [\![\mathfrak{d}_1 \cdots \mathfrak{d}_i]\!]$ for every $i > 0$ and thus $r \in [\![\mathfrak{d}_1\mathfrak{d}_2 \cdots]\!]$. By condition 1 we get that $\{r\} = [\![\mathfrak{d}_1\mathfrak{d}_2 \cdots]\!]$. ☐

In the motivation for IFS-representations, we said that they generalize both radix and LFT-representations. Now we can illustrate the idea formally for the signed binary representation and define it as an affine IFS-representation:

Example 3.2.6 (Signed binary representation).
The IFS-representation $\mathcal{S}_{\text{sigbin}} = ([-1, 1], \{\bar{1}, 0, 1\}, D_{\text{sigbin}})$ where

$$D_{\text{sigbin}}(\bar{1})(x) = \frac{x-1}{2} \qquad D_{\text{sigbin}}(0)(x) = \frac{x}{2} \qquad D_{\text{sigbin}}(1)(x) = \frac{x+1}{2}$$

is called the *signed binary representation* (see Figure 3.2).

The following lemma confirms that what we have just defined coincides with the traditional definition of the signed binary representation:

[2] The IFS-representations in practice often include some extra prefixes to extend the representation to the whole of the (extended) real space. We will not consider these prefixes in this thesis because a substantial part of a representing sequence in any IFS-representation suitable for finite computation always remains an IFS-representation in our sense.

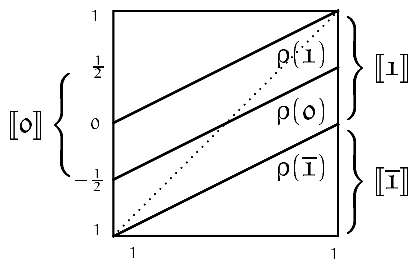

Figure 3.2: The d-IFS interpretation of the signed binary representation digits.

Lemma 3.2.7. *In the signed binary representation, it holds*

$$[\![\mathfrak{d}_1 \cdots \mathfrak{d}_n]\!] = \Big[\sum_{1\leqslant i\leqslant n} (2^{-i}\cdot \mathfrak{d}_i) - 2^{-n}\,,\ \sum_{1\leqslant i\leqslant n} (2^{-i}\cdot \mathfrak{d}_i) + 2^{-n}\Big] \tag{3.1}$$

$$[\![\mathfrak{d}_1\mathfrak{d}_2\mathfrak{d}_3\cdots]\!] = \left\{2^{-1}\cdot \mathfrak{d}_1 + 2^{-2}\cdot \mathfrak{d}_1 + 2^{-3}\cdot \mathfrak{d}_1 + \cdots\right\} \tag{3.2}$$

where the digits $\bar{1}, 0, 1$ *are interpreted as numbers* $-1, 0, 1$, *respectively.*

Proof. Let us proceed by induction on n. For $n = 0$, it holds

$$[\![\epsilon]\!] = I = [-1, 1] = [-2^0, +2^0].$$

Suppose that (3.1) holds for $n-1$.

$$\begin{aligned}
[\![\mathfrak{d}_1 \cdots \mathfrak{d}_n]\!] &= \left(\overline{D(\mathfrak{d}_1)} \circ \cdots \circ \overline{D(\mathfrak{d}_n)}\right)(I) = \overline{D(\mathfrak{d}_1)}\left([\![\mathfrak{d}_2\cdots\mathfrak{d}_n]\!]\right) = \\
&= \overline{D(\mathfrak{d}_1)}\Big(\Big[\sum_{1\leqslant i\leqslant n-1} (2^{-i}\cdot \mathfrak{d}_{i+1}) - 2^{-n+1}\,,\ \sum_{1\leqslant i\leqslant n-1} (2^{-i}\cdot \mathfrak{d}_{i+1}) + 2^{-n+1}\Big]\Big) = \\
&= \left[\frac{\sum_{i=1}^{n-1}(2^{-i}\cdot \mathfrak{d}_{i+1}) - 2^{-n+1} + \mathfrak{d}_1}{2}\,,\ \frac{\sum_{i=1}^{n-1}(2^{-i}\cdot \mathfrak{d}_{i+1}) + 2^{-n+1} + \mathfrak{d}_1}{2}\right] = \\
&= \Big[\sum_{1\leqslant i\leqslant n} (2^{-i}\cdot \mathfrak{d}_i) - 2^{-n}\,,\ \sum_{1\leqslant i\leqslant n} (2^{-i}\cdot \mathfrak{d}_i) + 2^{-n}\Big].
\end{aligned}$$

Equation (3.2) follows from (3.1) and from the definition of $[\![\cdot]\!]$ and the fact that a shrinking sequence of closed intervals converges to a point when its sizes converge to zero. □

3.2.1 Properties

The following is a relatively obvious property. Nevertheless, it is useful to point it out.

Lemma 3.2.8. *Let* (I, Σ, D) *be an IFS-representation. For any open interval* $J \subseteq I$, *there exists a sequence* $u \in \Sigma^*$ *such that* $[\![u]\!] \subset J$.

Proof. Take any point $r \in J$ and some of its representations $\alpha \in \Sigma^\omega$: $[\![\alpha]\!] = r$. Because J is an open neighbourhood of r and $\bigcap_{k\to\infty} [\![\alpha[1,k]]\!] = \{r\}$, for some k, it has to hold $[\![\alpha[1,k]]\!] \subset J$. Put $u = \alpha[1,k]$. □

Although d-contractions are not closed on composition in general, those that constitute an IFS-representation have to be:

Lemma 3.2.9 (Composition of digits).
In any IFS-representation (I, Σ, D)*, any composition* $D(\mathfrak{d}_1) \circ \cdots \circ D(\mathfrak{d}_n)$ *of the d-contractions of finitely many digits* $\mathfrak{d}_1, \ldots, \mathfrak{d}_n \in \Sigma$ *is a d-contraction.*

Proof. Denote $u = \mathfrak{d}_1 \cdots \mathfrak{d}_n$ and $f = D(\mathfrak{d}_1) \circ \cdots \circ D(\mathfrak{d}_n)$. We need to prove that f is a d-contraction. Since it is a composition of d-contractions, it is twice differentiable and its derivative is strictly positive. All that remains to be proved is that it has a unique fixpoint.

By condition 1 of Definition 3.2.4, there is a number $r \in I$ such that $[\![uuu \cdots]\!] = \bigcap_{i \in \mathbb{N}} f^i(I) = \{r\}$. It follows that r is a fixpoint of f. On the other hand, if x is a fixpoint of f, then $x \in f^i(I)$ for any $i \in \mathbb{N}$. This means that $x \in [\![uuu \cdots]\!]$, i.e. $x = r$. $\square$

Definition 3.2.10. For a sequence of digits $\mathfrak{d}_1 \cdots \mathfrak{d}_n \in \Sigma^*$ of an IFS-representation (I, Σ, D), we denote the d-contraction $D(\mathfrak{d}_1) \circ \cdots \circ D(\mathfrak{d}_n)$ by $\mathbf{D(\mathfrak{d}_1 \cdots \mathfrak{d}_n)}$.

Typically, there are many representations of a single number x. Sometimes, we need to pick a representation whose approximations contain a sub-sequence of a given sequence converging to x. By the following lemma, this is possible.

Lemma 3.2.11 (Representation along any sequence).
Let (I, Σ, D) *be an IFS-representation and* $x_k \in I$ *for* $k \in \mathbb{N}$*, a sequence of numbers converging to some* $x \in I$*. There exists a representation* $\alpha \in \Sigma^\omega$ *of* x *such that for every* $j \in \mathbb{N}$*, the approximation* $[\![\alpha[1, j]]\!]$ *contains* x_k *for some* $k \in \mathbb{N}$*.*

Proof. Construct the representation $\alpha = \mathfrak{d}_1\mathfrak{d}_2\ldots$ together with a sequence $y_0, y_1, \cdots \in I$ inductively as follows. Start with $y_0 := x$ and let $\mathfrak{d}_1 \in \Sigma$ be a digit whose semantics $[\![\mathfrak{d}_1]\!]$ covers x and at the same time x_k for infinitely many k. This is clearly possible if x is in the interior of $[\![\mathfrak{d}_1]\!]$ or if x is an endpoint of the base interval I because then $[\![\mathfrak{d}_1]\!]$ is a neighbourhood of x in I. If there is no such digit for $x \in I^o$, then there have to be two digits $\mathfrak{d}_1^L$ and $\mathfrak{d}_1^R$ in Σ such that x is the right hand side endpoint of $[\![\mathfrak{d}_1^L]\!]$ and the left hand side endpoint of $[\![\mathfrak{d}_1^R]\!]$. Therefore their union $[\![\mathfrak{d}_1^L]\!] \cup [\![\mathfrak{d}_1^R]\!]$ is a neighbourhood of x and contains x_k for infinitely many k. It holds either $x_k \in [\![\mathfrak{d}_1^L]\!]$ for infinitely many k or $x_k \in [\![\mathfrak{d}_1^R]\!]$ for infinitely many k. In the former case put $\mathfrak{d}_1 := \mathfrak{d}_1^L$, in the latter $\mathfrak{d}_1 := \mathfrak{d}_1^R$.

For every $j > 0$, put $y_j := D(\mathfrak{d}_j)^{-1}(y_{j-1})$ and let $\mathfrak{d}_j$ be the digit for which $y_j \in [\![\mathfrak{d}_j]\!]$ and at the same time $[\![\mathfrak{d}_1 \cdots \mathfrak{d}_j]\!]$ contains x_k for infinitely many k. The existence of such a $\mathfrak{d}_j$ can be guaranteed similarly like the existence of $\mathfrak{d}_1$ above but taking into account the fact that $x \in [\![\mathfrak{d}_1 \cdots \mathfrak{d}_j]\!]$ as well as $x_k \in [\![\mathfrak{d}_1 \cdots \mathfrak{d}_j]\!]$ for infinitely many k, for all $j \in \mathbb{N}$. $\square$

3.2.2 $\mathcal{S}$-adic numbers

Like dyadic numbers in (signed) binary representation, $\mathcal{S}$-adic numbers will be those that form the endpoints of an interval conveyed by a finite sequence of digits of a representation $\mathcal{S}$.

Definition 3.2.12. Let $\mathcal{S} = ([a, b], \Sigma, D)$ be an IFS-representation. The elements of the following set are called *(left/right) $\mathcal{S}$-adic numbers*:

$$\mathbf{Adic}_{\mathcal{S}} := \mathrm{LAdic}_{\mathcal{S}} \cup \mathrm{RAdic}_{\mathcal{S}} \qquad \text{where}$$
$$\mathbf{LAdic}_{\mathcal{S}} := \left\{ \big(D(u)\big)(a) \;\middle|\; u \in \Sigma^* \right\} \quad \text{and} \quad \mathbf{RAdic}_{\mathcal{S}} := \left\{ \big(D(u)\big)(b) \;\middle|\; u \in \Sigma^* \right\}$$

Define also the set of *$\mathcal{S}$-adic numbers of order less or equal ℓ*:

$$\mathbf{Adic}_{\mathcal{S}}^{\leqslant \ell} := \left\{ \big(D(u)\big)(a), \big(D(u)\big)(b) \;\middle|\; u \in \Sigma^{\ell} \right\}$$

for all $\ell \in \mathbb{N}$.

If $\mathcal{S}$ is the signed binary representation, the numbers in $\mathrm{Adic}_{\mathcal{S}}$ are indeed the dyadic numbers in $[-1, 1]$ exactly, while $\mathrm{LAdic}_{\mathcal{S}}$ and $\mathrm{RAdic}_{\mathcal{S}}$ consist of all the dyadic numbers in $[-1, 1)$ and $(-1, 1]$, respectively. Although it seems pointless to distinguish left and right dyadic numbers, for a general representation $\mathcal{S}$, this might not be the case as the sets $\mathrm{LAdic}_{\mathcal{S}}$ and $\mathrm{RAdic}_{\mathcal{S}}$ may be even disjoint.

Let us now prove lemmas which will help us to prove the continuity of many-valued functions expressible by a transducer in Theorem 3.3.8 on page 42.

Lemma 3.2.13. *For any IFS-representation $\mathcal{S} = (I, \Sigma, D)$, $x \in I$ and $\ell \in \mathbb{N}$, there exists $\varepsilon > 0$ such that $\big((x - \varepsilon, x) \cup (x, x + \varepsilon)\big) \cap \mathrm{Adic}_{\mathcal{S}}^{\leqslant \ell} = \emptyset$.*

Proof. The statement follows from the fact that $\mathrm{Adic}_{\mathcal{S}}^{\leqslant \ell}$ is a finite set. ☐

The following lemma expresses the intuitive idea that the closer two numbers are to each other, the closer representing sequences they can have.

Lemma 3.2.14. *Let $\mathcal{S} = (I, \Sigma, D)$ be an IFS-representation, $x, x' \in I$ with $x < x'$ (respective $x' < x$) and $(x, x'] \cap \mathrm{Adic}_{\mathcal{S}}^{\leqslant \ell} = \emptyset$ (respective $[x', x) \cap \mathrm{Adic}_{\mathcal{S}}^{\leqslant \ell} = \emptyset$) for some $\ell \in \mathbb{N}$. Then for any representation $\alpha' \in \Sigma^{\omega}$ of x', there is a representation α of x which agrees with α' on the first ℓ digits, i.e. $\alpha[1, \ell] = \alpha'[1, \ell]$.*

Proof. We need to show that $x \in [\![\alpha'[1, \ell]]\!]$. Suppose that $x \notin [\![\alpha'[1, \ell]]\!]$. Since $[\![\alpha'[1, \ell]]\!] = [D(\alpha'[1, \ell])(a), D(\alpha'[1, \ell])(b)]$ where $[a, b] := I$, the number $D(\alpha'[1, \ell])(a)$ is in $(x, x']$ (respective $D(\alpha'[1, \ell])(b)$ is in $[x', x)$). Both of these numbers are in $\mathrm{Adic}_{\mathcal{S}}^{\leqslant \ell}$ which is a contradiction. ☐

3.2.3 Open IFS-representations

We have mentioned in the introduction the classical observation that an admissible representation has to be redundant. Rather than defining redundancy, in our context, it turns out to be more fitting to state this condition in terms of interior overlapping. The term 'open' refers to the fact that every interior point of the base interval can be covered by the interior of a digit range.

Definition 3.2.15 (Open IFS-representation).
An IFS-representation (I, Σ, D) is called *open* if for each $r \in I^o$ (i.e. in the interior of I), there exists a digit $\mathfrak{d} \in \Sigma$ such that $r \in [\![\mathfrak{d}]\!]^o$.

The crucial property of open representations is that it can represent every number by an 'open' representation in the following sense:

Definition 3.2.16 (Open approximation).
In an IFS-representation (I, Σ, D), a representing sequence $\alpha \in \Sigma^\omega$ is called an *open representation* of r if $r \in [\![\alpha[1, k]]\!]^o$ for every $k \in \mathbb{N}$.

Lemma 3.2.17. *In an open IFS-representation* (I, Σ, D) *for every number* $r \in I^o$*, there is an open representation of* r*.*

Proof. In a similar fashion like in the proof of Lemma 3.2.5, define $\alpha := \mathfrak{d}_1\mathfrak{d}_2\cdots$ inductively together with a sequence $r_1, r_2, \ldots \in I^o$. First, set $r_1 = r$ and let $\mathfrak{d}_i$ be a digit for which $r_i \in [\![\mathfrak{d}_i]\!]^o$ and put $r_{i+1} := D(\mathfrak{d}_i)^{-1}(r_i)$. Since $r_i \in [\![\mathfrak{d}_i]\!]^o$, it holds $r_{i+1} \in I^o$ because $D(\mathfrak{d}_i)$ is an increasing function. Now we can prove that α has the desired property because it holds

$$r = \big(D(\mathfrak{d}_1) \circ \cdots \circ D(\mathfrak{d}_i)\big)(r_{i+1}) = D(\alpha[1, i])(r_{i+1}).$$

Since $r_{i+1} \in I^o$ and $D(\alpha[1, i])$ is an increasing function, also the image of r_{i+1} by $D(\alpha[1, i])$ is in the interior of the range, i.e. $r \in [\![\alpha[1, i]]\!]^o$. $\square$

We prove the following obvious statement only to illustrate a general proof technique for showing that something is an open IFS-representation.

Lemma 3.2.18. *The signed binary representation is an open IFS-representation.*

Proof. It follows from the proof of Lemma 3.2.7 that the digits of the signed binary representation contract strongly enough (cf. Definition 3.2.4 point (1)). We need to observe that every number $r \in [-1, 1]^o$ is in the interior of the domain of some of the digits. Since these interiors are the open intervals $(-1, 0)$, $(-\frac{1}{2}, \frac{1}{2})$ and $(0, 1)$, this condition is true. The rest follows from Lemma 3.2.5. $\square$

Figure 3.3 shows how several initial digits in the signed binary representation produce interval approximations for the represented number by means of composition of their d-contractions.

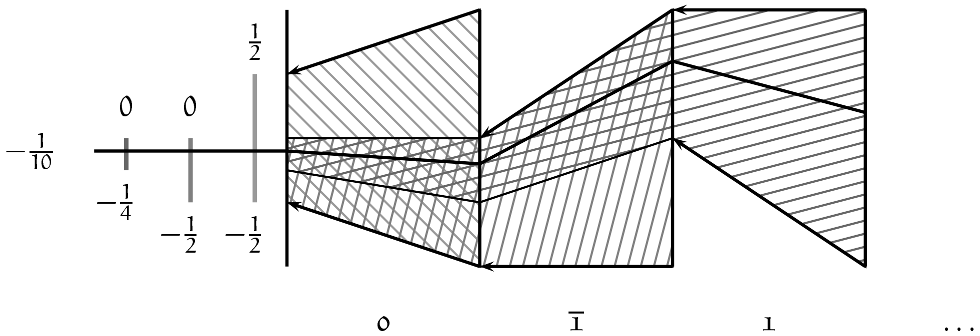

Figure 3.3: Representing $-1/10$ in the signed binary representation.

3.3 Real number IFS-transducers

3.3.1 Stream transducers

Any discrete computing system can be viewed as some kind of a state transition diagram. The one in the following definition describes computation with a finite number of input streams and one output stream. Moreover, it is non-deterministic in a very liberal way. The only constraint, even though a realistic one, is that only finitely many transitions are possible from each state.

Definition 3.3.1.

1. A *stream transducer* of type

$$(\boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_n) \rightsquigarrow \boldsymbol{\Sigma}_0$$

   is a tuple $(\mathbf{Q}, \zeta, \mathbf{q_I})$ consisting of

   – a set Q of *states* including the *initial state* $q_I \in Q$

   – a non-deterministic *transition function*

$$\zeta \colon Q \to \wp_{\text{fin}}(Q \times \Sigma_1^* \times \cdots \times \Sigma_n^* \times \Sigma_0^*)$$

   assigning a *finite* set of transitions to each state. Each of these transitions consists of a target state and a string of digits for each channel to be passed along it.

2. A stream transducer which has a finite number of states is called a *finite (stream) transducer*.

3. For a stream transducer $\mathcal{T} = (Q, \zeta, q_I)$, we name the following domain of sequences for the purposes of this definition:

$$E := \left(Q \cup \{(c, s) \mid c \in \{0, 1, \dots, n\}, s \in \Sigma_c^*\}\right)^\infty$$

   We define the *history set* consisting of *computations* of $\mathcal{T}$, denoted by $\mathcal{H}_{\mathcal{T}}^{\mathbf{fin}}$ to be the minimal subset of E satisfying:

- $q_I \in \mathcal{H}^{\text{fin}}_{\mathcal{T}}$ and
- $u \cdot q \in \mathcal{H}^{\text{fin}}_{\mathcal{T}}$ for some $q \in Q$ implies that also

$$u \cdot q \cdot (1, s_1) \cdots (n, s_n) \cdot (0, s_0) \cdot q' \in \mathcal{H}^{\text{fin}}_{\mathcal{T}}$$

for any $(q', s_1, \dots, s_n, s_0) \in \zeta(q)$.

Let $\boldsymbol{\mathcal{H}_{\mathcal{T}}}$ be the Scott-closure of $\mathcal{H}^{\text{fin}}_{\mathcal{T}}$ within the prefix ordered domain of sequences, i.e. $\mathcal{H}_{\mathcal{T}}$ is $\mathcal{H}^{\text{fin}}_{\mathcal{T}}$ enriched by the infinite sequences γ with $(\forall k)(\exists n > k)(\gamma[1, n] \in \mathcal{H}^{\text{fin}}_{\mathcal{T}})$.

4. For a subset of channel indices $P \subseteq \{0, 1, \dots, n\}$, we define the *projection to channels* P of a set $H \subseteq E$ as $\mathbf{H}|_P := \{\gamma|_P \mid \gamma \in H\}$ where $\_|_P$ is a continuous stream operator defined for prefixes of sequences from H as follows:

   - $\epsilon|_P = \epsilon$,
   - $(u \cdot q)|_P = u|_P$ for any $q \in Q$,
   - $(u \cdot (c, s))|_P = \begin{cases} u|_P \cdot (c, s) & \text{if } c \in P \\ u|_P & \text{otherwise} \end{cases}$

   For $P = \{i\}$, we can write $\mathbf{H}|_{\mathbf{i}}$ instead of $H|_{\{i\}}$.
   We also use special notation **In** $:= \{1, \dots, n\}$ and **Out** $:= \{0\}$.

5. The sequence $\gamma|_i$ for some $\gamma \in H$ is of the form $(i, s_1)(i, s_2)\cdots$. In order to remove the redundant channel numbers i we define $\boldsymbol{\Pi_i(\gamma)} := s_1 s_2 \cdots$. Similarly a $\gamma|_P$ with $|P| > 1$ can be turned into an $|P|$-tuple of sequences sorted according to the channel numbers: $\boldsymbol{\Pi_P(\gamma)} := \big(\Pi_i(\gamma)\big)_{i \in P}$. On the level of sets we define $\boldsymbol{\Pi_P(H)} := \big\{\Pi_P(\gamma) \;\big|\; \gamma \in H\big\}$.

6. For every finite sequence $\gamma \in \mathcal{H}^{\text{fin}}_{\mathcal{T}}$, we denote by $\boldsymbol{q_\gamma}$ the state that appears at its end.

7. For any tuple of sequences $\vec{\alpha} = (\alpha_1, \dots, \alpha_n) \in \Sigma_1^\infty \times \cdots \times \Sigma_n^\infty$, we define $\boldsymbol{\mathcal{H}_{\mathcal{T}}(\vec{\alpha})}$ to be the set of the maximal sequences in $\mathcal{H}_{\mathcal{T}}$ that follow $\vec{\alpha}$ on the input channels, as follows:

$$\mathcal{H}_{\mathcal{T}}(\vec{\alpha}) := \max\Big\{\gamma \in \mathcal{H}_{\mathcal{T}} \;\Big|\; \big(\forall i \in \{1, \dots, n\}\big)\big(\Pi_i(\gamma) \sqsubseteq \alpha_i\big)\Big\}.$$

   Notice that $\mathcal{H}_{\mathcal{T}}(\vec{\alpha})$ is never empty because the set on the right hand side contains at least the sequence q_I.

8. A transducer $\mathcal{T} = (Q, \zeta, q_I)$ is said to *compute* the many-valued stream function

$$\boldsymbol{\sigma_{\mathcal{T}}}\colon \Sigma_1^\infty \times \cdots \times \Sigma_n^\infty \rightrightarrows \Sigma_0^\infty : \vec{\alpha} \mapsto \Pi_{\text{Out}}\left(\mathcal{H}_{\mathcal{T}}(\vec{\alpha})\right)$$

9. A transducer $\mathcal{T}$ is called

   - *weakly deterministic* if $\sigma_{\mathcal{T}}$ is single-valued,
   - *deterministic* if $\mathcal{H}_{\mathcal{T}}(\alpha)$ is a singleton set for each $\alpha \in \Sigma_1^\infty \times \cdots \times \Sigma_n^\infty$,

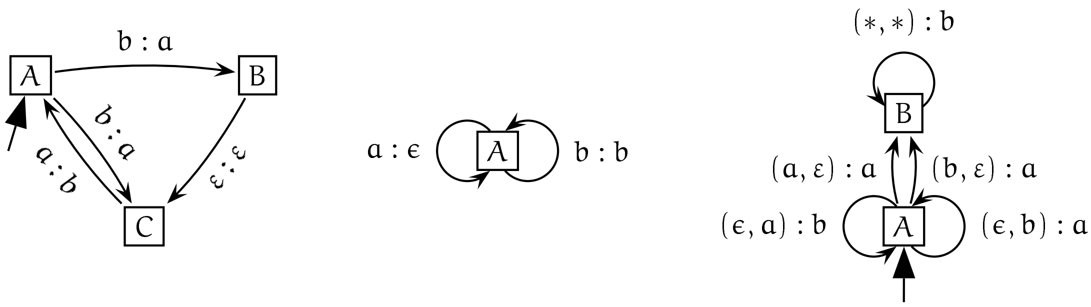

Figure 3.4: Simple examples of transducers: weakly deterministic, not talkative and ignorant.

- *total* if for every infinite $\vec{\alpha} \in \Sigma_1^\omega \times \cdots \times \Sigma_n^\omega$ also every $\gamma \in \mathcal{H}_{\mathcal{T}}(\vec{\alpha})$ is infinite (i.e. the transducer cannot get stuck except due to a lack of input),
- *talkative* if every infinite computation $\gamma \in \mathcal{H}_{\mathcal{T}}$ produces an infinite output $\Pi_{\mathrm{Out}}(\gamma)$,
- *ignorant* if there exists an infinite computation $\gamma \in \mathcal{H}_{\mathcal{T}}$ with a finite input $\Pi_i(\gamma)$ for some $i \in \mathrm{In}$.

Notice that a talkative transducer has to be also total. The three transducers in Figure 3.4 illustrate the notions which have just been defined. The first one is weakly deterministic but not deterministic, the second one is total but not talkative and the third one is talkative and deterministic but at the same time ignorant on the first input channel.

We have just used an intuitive way to visualize transducers as graphs in Figure 3.4. Let us formalize the idea.

Definition 3.3.2 (Underlying graph).
A transducer $\mathcal{T} = (Q, \zeta, q_I)$ naturally corresponds to an (oriented, pointed, edge labelled) graph $\mathcal{G}_{\mathcal{T}}$ (called the *underlying graph* of $\mathcal{T}$) whose vertices are the states Q, its root is the initial state, and there is an edge from q to q' labelled by a colon separated pair $(s_1, \ldots, s_n) : s_0$ for each $q \in Q$ and $(q', s_1, \ldots, s_n, s_0) \in \zeta(q)$.

A finite stream transducer can be even fully specified simply by drawing its underlying graph. It is also easy to check talkativity of finite transducers by observing their underlying graphs:

Lemma 3.3.3 (Detecting talkative finite transducers).
A finite transducer is talkative if and only if every reachable cycle in its underlying graph contains an edge with non-empty output.

Proof. Let $\mathcal{T}$ be a finite transducer of type $(\Sigma_1, \ldots, \Sigma_n) \rightrightarrows \Sigma_0$.

Assume first that $\mathcal{T}$ is talkative and suppose that there would be a computation $\gamma_1, \gamma_1\gamma_2 \in \mathcal{H}_{\mathcal{T}}^{\mathrm{fin}}$ with $\gamma_2 \neq \epsilon$ such that $q_{\gamma_1} = q_{\gamma_1\gamma_2}$ (a reachable cycle) with $\Pi_{\mathrm{Out}}(\gamma_2) = \epsilon$ (empty output). Then $\gamma_1 \cdot \gamma_2^\omega$ would be a computation with infinite input (which might or might not be consumed)

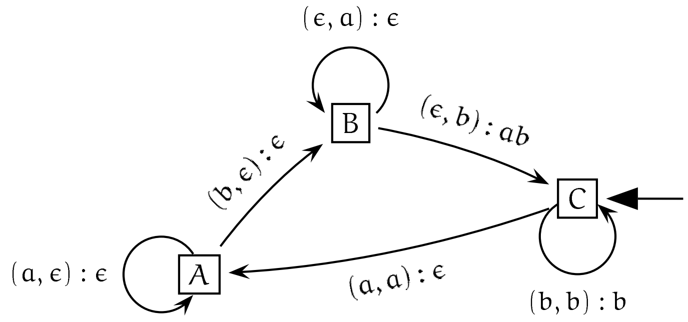

Figure 3.5: A finite transducer of the type $(\{a, b\}, \{a, b\}) \rightsquigarrow \{a, b\}$

that produces a finite output $\Pi_{\text{Out}}(\gamma_1)$ which contradicts the definition of $\mathcal{T}$ being talkative. This concludes the proof of the 'only if' part of the lemma.

Assume now that $\mathcal{T}$ has no reachable cycle with empty output and suppose that $\mathcal{T}$ is not talkative. This would mean that there would be an infinite computation $\gamma \in \mathcal{H}_{\mathcal{T}}$ with finite output u. It would be possible to find a finite prefix $\gamma_1 \in \mathcal{H}_{\mathcal{T}}^{\text{fin}}$ of γ such that $\Pi_{\text{Out}}(\gamma_1) = u$ and $\Pi_{\text{Out}}(\gamma') = \epsilon$ for $\gamma' := \gamma \setminus \gamma_1$. Among the infinitely many states in γ', some state has to appear infinitely many times. Therefore one can split γ' into three pieces: $\gamma' =: \gamma_2\gamma_3\gamma''$ with $\gamma_3 \neq \epsilon$ so that $q_{\gamma_1\gamma_2} = q_{\gamma_1\gamma_2\gamma_3}$. This means that there is a cycle leading through the reachable state $q_{\gamma_1\gamma_2}$ with empty total output $\Pi_{\text{Out}}(\gamma_3)$, which is a contradiction. ☐

Example 3.3.4. The transducer in figure Figure 3.5 is deterministic, neither total nor talkative and the function that it computes can be described as follows:

$$\sigma_{\mathcal{T}}(u_1, u_2) = \max\left\{u \;\middle|\; u \sqsubseteq \mathit{can}_{a=aa}(u_1) \;\&\; u \sqsubseteq \mathit{can}_{a=aa}(u_2)\right\}$$

where $\mathit{can}_{a=aa}\colon \{a, b\}^\infty \to \{a, b\}^\infty$ is the Scott-continuous function which assigns to each finite sequence u the shortest representative of its class in the semigroup congruence generated by the equality $a = aa$:

$$\mathit{can}_{a=aa}(a^{k_0}ba^{k_1}b \cdots ba^{k_n}) := a^{\min(1,k_0)}ba^{\min(1,k_1)} \cdots ba^{\min(1,k_n)}$$

for all $n, k_0, \ldots, k_n \in \mathbb{N}$.

3.3.2 Transducers modulo representations

In the previous subsection, it has been described how to formalize computation with infinite streams of symbols without regard to the meaning of the symbols and their sequences. From now on, we will consider only transducers that operate with streams of digits of some IFS-representations.

Many transducers, like Example 3.3.4, may use the alphabets which correspond to certain IFS-representations but do not respect the representation in either of several aspects:

A. An output does not need to be infinite, i.e. it does not represent a single real number, even if the sources of input digits are unlimited.

B1. An infinite input can lead to two or more infinite outputs each of which representing a different real number.

B2. Different representations of the same tuple of input real numbers may lead to outputs that represent different real numbers.

We will not put any extra requirements on a stream transducer to be qualified as an IFS-transducer. Therefore any of the situations A, B1, B2 can happen. But we will use various adjectives to indicate that a transducer respects the representations to some extent. A transducer for which A never happens will be called *precise*. If a transducer is not total, i.e. it stalls for some input, or it is not talkative, i.e. it can work silently forever without any output, then A inevitably happens, i.e. the transducer is not precise. On the other hand, a transducer which is talkative and total is precise. The negation of 'precise' is '*imprecise*'.

Transducers that avoid both B1 and B2 will be called *single-valued* as opposed to *many-valued*. Note that conditions B1 and B2 talk only about infinite inputs and outputs. Thus a single-valued machine can output several finite outputs with different meanings for the same incomplete input. (By incomplete input we mean a tuple of digit streams among which at least one is finite.) Even if this does not happen, the machine can still output several finite outputs with different meanings for several different incomplete inputs that represent the same tuple of incomplete values (i.e. intervals).

The following definitions formalize what we have just discussed.

Definition 3.3.5. An *IFS-transducer* of type $(\mathcal{S}_1, \ldots, \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ where $\mathcal{S}_0, \ldots, \mathcal{S}_n$ are IFS-representations, is a stream transducer of type $(\Sigma_{\mathcal{S}_1}, \ldots, \Sigma_{\mathcal{S}_n}) \rightsquigarrow \Sigma_{\mathcal{S}_0}$.

An IFS-transducer $\mathcal{T}$ is said to *completely compute the many-valued number to interval function*

$$\overline{f}_{\mathcal{T}}\colon I_{\mathcal{S}_1} \times \cdots \times I_{\mathcal{S}_n} \rightrightarrows \mathcal{I}(I_{\mathcal{S}_0})$$

defined to be the following factor of $\sigma_{\mathcal{T}}$ by the representation functions:

$$\overline{f}_{\mathcal{T}}(x_1, \ldots, x_n) := \left[\!\!\left[\sigma_{\mathcal{T}} \left([\![\{x_1\}]\!]_{\mathcal{S}_1}^{-1}, \ldots, [\![\{x_n\}]\!]_{\mathcal{S}_n}^{-1} \right) \right]\!\!\right]_{\mathcal{S}_0}$$

(In this formula, the functions $[\![\cdot]\!]_{\mathcal{S}_i}$ are naturally extended to sets and therefore each $[\![\{x_i\}]\!]_{\mathcal{S}_i}^{-1}$ is the set of all sequences that represent the number x_i. Similarly, $\sigma_{\mathcal{T}}$ is extended so that it operates on sets of sequences rather than on sequences.)

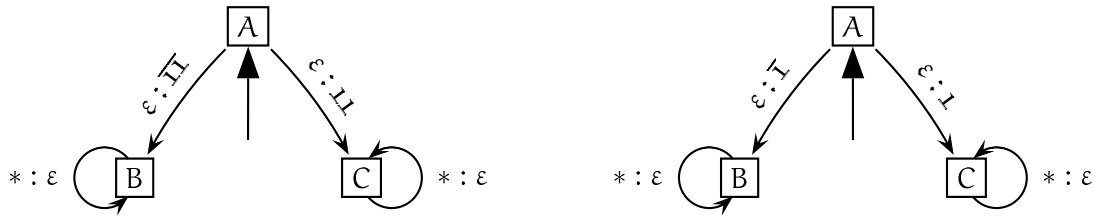

Figure 3.6: Non-precise total transducers—one is single valued, one is not.

Thus the function $\overline{f}_{\mathcal{T}}$ maps tuples of real numbers to sets of intervals. For example, the first transducer in Figure 3.6 maps the set $\{[-1,-1/2],[1/2,1]\}$ to any input real number from $[-1,1]$.

Whenever we will talk about a transducer $\mathcal{T}\colon (\mathcal{S}_1,\dots,\mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ in a context where it cannot be confused with another transducer, we adopt the following short notation: $I_i := I_{\mathcal{S}_i}$, $D_i := D_{\mathcal{S}_i}$ for all $i = 0,\dots,n$, and $V := I_1 \times \cdots \times I_n$ as usual.

Definition 3.3.6. An IFS-transducer $\mathcal{T}$ of type $(\mathcal{S}_1,\dots,\mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ is called

- *precise* if for each tuple of real numbers $\vec{x} \in V$, the set $\overline{f}_{\mathcal{T}}(\vec{x})$ contains only singletons.
- *single-valued* if for each tuple of real numbers $\vec{x} \in V$, the intersection of the members of the set $\overline{f}_{\mathcal{T}}(\vec{x})$ is not empty.
- *extensional* if it is precise and single-valued, i.e. the set $\overline{f}(\vec{x})$ contains exactly one singleton for each $\vec{x} \in V$.

There are examples of precise, yet not single-valued transducers in Figure 3.9 on page 48. On the other hand, there are transducers which are single-valued although not precise (see Figure 3.6 on the right). For examples of extensional transducers, see e.g. Figure 3.8.

The fact that $\mathcal{T}$ has some of the attributes above, can be expressed by assigning to it also one of the following types:

$$\textit{precise: } (\boldsymbol{\mathcal{S}}_1,\dots,\boldsymbol{\mathcal{S}}_n) \rightrightarrows \boldsymbol{\mathcal{S}}_0$$
$$\textit{extensional: } (\boldsymbol{\mathcal{S}}_1,\dots,\boldsymbol{\mathcal{S}}_n) \to \boldsymbol{\mathcal{S}}_0$$

A precise IFS-transducer $\mathcal{T}$ is said to completely compute the many-valued real number function

$$f_{\mathcal{T}}\colon I_{\mathcal{S}_1} \times \cdots \times I_{\mathcal{S}_n} \rightrightarrows I_{\mathcal{S}_0}$$

which is the restriction of $\overline{f}_{\mathcal{T}}$ by identifying each real number $x \in I_{\mathcal{S}_0}$ with the singleton set $\{x\}$.

If $\mathcal{T}$ is extensional, the function $f_{\mathcal{T}}$ is an ordinary single-valued function of type

$$f_{\mathcal{T}}\colon I_{\mathcal{S}_1} \times \cdots \times I_{\mathcal{S}_n} \to I_{\mathcal{S}_0}.$$

Any function $f \supseteq f_{\mathcal{T}}$ is said to be *computed by* $\mathcal{T}$ for a precise transducer $\mathcal{T}$.

Notice that a single-valued function is computed if and only if it is completely computed by a transducer. Therefore for single-valued functions and extensional transducers, we do not need to distinguish the two notions of 'being computed'.

Remark 3.3.7. An IFS-transducer is precise if and only if it is total and talkative.

The following theorem states a transducer version of the famous theorem which says that computability entails continuity.

Theorem 3.3.8. *The function completely computed by a precise IFS-transducer* $\mathcal{T}\colon (\mathcal{S}_1, \dots, \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ *is continuous.*

Proof. Let $f := f_{\mathcal{T}}$. Take an arbitrary open set $U \subseteq I_0$ and put $E := \{\vec{x} \in V \mid f(\vec{x}) \subseteq U\}$. We aim at proving that E is open by showing that an arbitrary $\vec{x} \in E$ has some neighbourhood in E.

Consider an arbitrary infinite computation $\gamma \in \mathcal{H}_{\mathcal{T}}$ with $[\![\Pi_{\mathrm{In}}(\gamma)]\!] \ni \vec{x}$. (There is one because $\mathcal{T}$ is total.) Since $[\![\Pi_{\mathrm{Out}}(\gamma)]\!] \in f(x) \subseteq U$ and U is open, there is a finite prefix $\gamma_U \sqsubseteq \gamma$, $\gamma_U \in \mathcal{H}_{\mathcal{T}}^{\mathrm{fin}}$, such that $[\![\Pi_{\mathrm{Out}}(\gamma_U)]\!] \subset U$. Moreover, let γ_U be the shortest among the prefixes with this property.

Define $G := \{\gamma_U \mid \gamma \in \mathcal{H}_{\mathcal{T}}, \gamma \text{ is infinite}, [\![\Pi_{\mathrm{In}}(\gamma)]\!] \ni \vec{x}\}$. With the prefix order, the prefix closure of G is a partial order whose Hasse diagram is a tree. This tree is finitely branching and there is no infinite increasing sequence. By König's lemma the tree is finite and thus also G has to be finite.

Let $\ell := \max\big\{|\Pi_i(\gamma_U)| \;\big|\; \gamma_U \in G, i \in \{1, \dots, n\}\big\}$ be the maximum length of input for the computations within G. Take such an $\varepsilon > 0$ for which the union $(x_i - \varepsilon, x_i) \cup (x_i, x_i + \varepsilon)$ does not contain any $\mathcal{S}_i$-adic number of order less or equal to ℓ (by Lemma 3.2.13) for all $i = 1, \dots, n$.

Now for any representation $\vec{\alpha}'$ of any $\vec{x}' \in (\vec{x} - \varepsilon, \vec{x} + \varepsilon)$, there is a representation $\vec{\alpha}$ of $\vec{x}$ with $\alpha_i[1, \ell] = \alpha'_i[1, \ell]$ for all $i = 1, \dots, n$ by Lemma 3.2.14. Using the assumption that $\mathcal{T}$ is total, for any computation γ' with $\Pi_{\mathrm{In}}(\gamma') = \vec{x}'$, there is a computation $\gamma_U \in G$ with $\gamma_U \sqsubseteq \gamma'$. Thus the output of γ' is in U: $[\![\Pi_{\mathrm{Out}}(\gamma')]\!] \in [\![\Pi_{\mathrm{Out}}(\gamma_U)]\!] \subset U$.

We proved that any computation γ' of $\mathcal{T}$ with an input $\vec{x}' \in (\vec{x} - \varepsilon, \vec{x} + \varepsilon)$ produces output in U. This means $f(\vec{x}') \subseteq U$ for all $\vec{x}' \in (\vec{x} - \varepsilon, \vec{x} + \varepsilon)$, i.e. $(\vec{x} - \varepsilon, \vec{x} + \varepsilon) \subseteq E$ which concludes the proof that E is open. $\square$

3.3.3 Composing finite transducers

By the intuitive understanding, transducers are devices that use a finite number of input and output one-way channels as a sole means of communication with the outside world. Using this idea, one can easily imagine two finite transducers next to each other and connecting the output channel of one with an input channel of the other one, provided that the two channels use the same representation. It seems quite obvious that this new unit would still be a finite state transducer completely computing the composition of the functions completely computed by the original machines. We will formally express this intuition and check that it is correct.

Although we will not make any use of it, we should note the fact that the intuitive idea of *splitting a channel* into two and connecting it to two input channels of another transducer is possible only with a buffer. This means that channel splitting does not preserve finiteness in general. The same is true for *loopback*—i.e. splitting the output channel and connecting one branch to an input channel of the same transducer. And it is also true for compositions of transducers with multiple output channels by connecting more than one channel between the two such transducers.

Definition 3.3.9. Let $\mathcal{T}_1 = (Q_1, \zeta_1, (q_I)_1)$ and $\mathcal{T}_2 = (Q_2, \zeta_2, (q_I)_2)$ be *finite* precise transducers with the following types:

$$\mathcal{T}_1\colon (\mathcal{S}_1, \dots, \mathcal{S}_n) \rightrightarrows \mathcal{S}_0,$$
$$\mathcal{T}_2\colon (\mathcal{S}'_1, \dots, \mathcal{S}'_m) \rightrightarrows \mathcal{S}'_0,$$

and let $\mathcal{S}_0 = \mathcal{S}'_k$ for some $1 \leqslant k \leqslant m$. The tuple (Q, ζ, q_I) as defined below, is denoted by $\mathcal{T}_2[k \leftarrow \mathcal{T}_1]$ and called the *composition of* $\mathcal{T}_2$ *and* $\mathcal{T}_1$ *on channel* k.

The construction of $\mathcal{T}_2[k \leftarrow \mathcal{T}_1]$:

- $Q = Q_1 \times Q_2 \times \Sigma_{\mathcal{S}_k}^{L_1+L_2}$
  where L_1 is the maximum length of an output during a single transition of $\mathcal{T}_1$ and L_2 is the maximum length of an input on channel k in a single transition of $\mathcal{T}_2$:

  $$L_1 = \max\left\{|s_0| \,\middle|\, (q', s_1, \dots, s_n, s_0) \in \zeta_1(q)\right\}$$
  $$L_2 = \max\left\{|s'_k| \,\middle|\, (q', s'_1, \dots, s'_n, s'_0) \in \zeta_2(q)\right\}$$

  (The $\Sigma_{\mathcal{S}_k}^{L_1+L_2}$ portion of a state is the contents of a finite buffer used for storing the output of $\mathcal{T}_1$ before it is consumed by $\mathcal{T}_2$ on the k-th channel.)

- $q_I = \big((q_I)_1, (q_I)_2, \epsilon\big)$
  (In the beginning, the buffer is empty.)

- $\zeta\big((q_1, q_2, s)\big) =$

  $$\left\{\big((q'_1, q_2, s \cdot s_0), \underbrace{\epsilon, \dots, \epsilon}_{(k-1)\times}, s_1, \dots, s_n, \underbrace{\epsilon, \dots, \epsilon}_{(m-k)\times}, \epsilon\big) \,\middle|\, |s| < L_2, (q'_1, s_1, \dots, s_n, s_0) \in \zeta_1(q_1)\right\} \cup$$

  ($\mathcal{T}_1$ moves)

  $$\left\{\big((q_1, q'_2, \overline{s}), s'_1, \dots, s'_{k-1}, \underbrace{\epsilon, \dots, \epsilon}_{n\times}, s'_{k+1}, \dots, s'_m, s'_0\big) \,\middle|\, s = s'_k\overline{s}, (q'_2, s'_1, \dots, s'_m, s'_0) \in \zeta_2(q_2)\right\}$$

  ($\mathcal{T}_2$ moves)

Proposition 3.3.10. *The tuple* $\mathcal{T} := (Q, \zeta, q_I)$ *from the previous definition is a finite precise IFS-transducer of type*

$$(\mathcal{S}'_1, \dots, \mathcal{S}'_{k-1}, \mathcal{S}_1, \dots, \mathcal{S}_n, \mathcal{S}'_{k+1}, \dots, \mathcal{S}'_m) \rightrightarrows \mathcal{S}'_0.$$

and it holds

$$f_{\mathcal{T}}(x_1, \dots, x_{n+m-1}) = f_{\mathcal{T}_2}(x_1, \dots, x_{k-1}, f_{\mathcal{T}_1}(x_k, \dots, x_{k+n-1}), x_{k+n}, \dots, x_{n+m-1}) \tag{3.3}$$

for all $(x_1, \dots, x_{n+m-1}) \in I_{\mathcal{S}'_1} \times \dots \times I_{\mathcal{S}'_{k-1}} \times I_{\mathcal{S}_1} \times \dots \times I_{\mathcal{S}_n} \times I_{\mathcal{S}'_{k+1}} \times \dots \times I_{\mathcal{S}'_m}$.
If both $\mathcal{T}_1$ *and* $\mathcal{T}_2$ *are extensional, also* $\mathcal{T}$ *is extensional.*

Proof outline. The proof of the first part of the proposition amounts to type-checking of the construction which is trivial in this case.

The equation (3.3) follows from its stream-level analogue:

$$\sigma_{\mathcal{T}}(\alpha_1, \dots, \alpha_{n+m-1}) = \sigma_{\mathcal{T}_2}(\alpha_1, \dots, \alpha_{k-1}, \sigma_{\mathcal{T}_1}(\alpha_k, \dots, \alpha_{k+n-1}), \alpha_{k+n}, \dots, \alpha_{n+m-1})$$

for any $(\alpha_1, \dots, \alpha_{n+m}) \in \Sigma^{\infty}_{\mathcal{S}'_1} \times \dots \times \Sigma^{\infty}_{\mathcal{S}'_{k-1}} \times \Sigma^{\infty}_{\mathcal{S}_1} \times \dots \times \Sigma^{\infty}_{\mathcal{S}_n} \times \Sigma^{\infty}_{\mathcal{S}'_{k+1}} \times \dots \times \Sigma^{\infty}_{\mathcal{S}'_m}$. This, in turn, results from the following property: To each computation $\gamma \in \mathcal{H}_{\mathcal{T}}$ there corresponds a pair of computations $\gamma_1 \in \mathcal{H}_{\mathcal{T}_1}$ and $\gamma_2 \in \mathcal{H}_{\mathcal{T}_2}$ with

$$\Pi_k(\gamma_2) \sqsubseteq \Pi_{\mathrm{Out}}(\gamma_1) \quad \text{and} \quad \left|\Pi_{\mathrm{Out}}(\gamma_1) \setminus \Pi_k(\gamma_2)\right| \leqslant L_1 + L_2 \tag{3.4}$$

such that

$$\begin{aligned} &\Pi_{\mathrm{Out}}(\gamma) = \Pi_{\mathrm{Out}}(\gamma_2) \quad \text{and} \\ &\Pi_{\mathrm{In}}(\gamma) = \big(\Pi_1(\gamma_2), \dots, \Pi_{k-1}(\gamma_2), \Pi_1(\gamma_1), \dots, \Pi_n(\gamma_1), \Pi_{k+1}(\gamma_2), \dots, \Pi_m(\gamma_2)\big) \end{aligned} \tag{3.5}$$

and vice versa, i.e. for any $\gamma_1 \in \mathcal{H}_{\mathcal{T}_1}$ and $\gamma_2 \in \mathcal{H}_{\mathcal{T}_2}$ with (3.4), there is $\gamma \in \mathcal{H}_{\mathcal{T}}$ such that (3.5) holds. This correspondence can be proved by induction on the lengths of γ, γ_1 and γ_2.

3.3.4 Finitely computable numbers

Finitely computable real numbers (shortly finite numbers) are very easy to characterize. It was already mentioned in the introduction that they are the numbers that have a periodic representation. We define them formally because we need some of their properties in the following chapter.

Definition 3.3.11 (Finite numbers).
Let $\mathcal{S} = (I, \Sigma, D)$ be an IFS-representation. The set of all constants $c \in I$ for which there is a finite extensional transducer $\mathcal{T}$ of type $() \to \mathcal{S}$ with $f_{\mathcal{T}} = c$, is denoted by $\mathbf{Fin}_{\mathcal{S}}$.

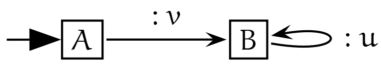

Figure 3.7: A transducer computing $[\![vuuu\dots]\!]$.

Lemma 3.3.12 (Finite numbers are periodically representable).
For any $x \in \mathrm{Fin}_{\mathcal{S}}$ *(for some IFS-representation* $\mathcal{S}$*), there are* $u, v \in \Sigma_{\mathcal{S}}^*$ *such that* $x = [\![vuuu\dots]\!]$*. On the other hand, for any* $v, u \in \Sigma_{\mathcal{S}}^*$*, it holds* $[\![vuuu\dots]\!] \in \mathrm{Fin}_{\mathcal{S}}$.

Proof. The first part follows from a well known fact that deterministic finite transducers without input can generate only periodical sequences (see, for example, [Trakhtenbrot & Barzdin 1973]). Although our machines can be non-deterministic, they can be made deterministic by removing some transitions while retaining an output which is correct modulo representation. On the other hand, the number $[\![vuuu\dots]\!]$ can be computed by the two state transducer shown in Figure 3.7. □

Lemma 3.3.13 (Finite numbers are dense).
For any IFS-representation $\mathcal{S}$*, the set* $\mathrm{Fin}_{\mathcal{S}}$ *is dense in* $I_{\mathcal{S}}$.

Proof. Choose any open set $U \subset I_{\mathcal{S}}$ and by Lemma 3.2.8, get $u \in \Sigma_{\mathcal{S}}^*$ with $[\![u]\!] \subset U$. The finitely computable number $[\![uuu\dots]\!]$ is in U. □

The following lemma is a special case of Proposition 3.3.10.

Lemma 3.3.14 (Instantiating an argument).
For any precise finite transducer $\mathcal{T}$ *of type* $(\mathcal{S}_1, \dots, \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ *any index* $1 \leqslant k \leqslant n$ *and a finitely computable number* $x \in \mathrm{Fin}_{\mathcal{S}_k}$*, there is a finite transducer* $\mathcal{T}'$ *of type* $(\mathcal{S}_1, \dots, \mathcal{S}_{k-1}, \mathcal{S}_{k+1}, \dots, \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$*, completely computing the function*

$$f_{\mathcal{T}'}(x_1, \dots, x_{n-1}) = f_{\mathcal{T}}(x_1, \dots, x_{k-1}, x, x_k, \dots, x_{n-1}) \tag{3.6}$$

i.e. $f_{\mathcal{T}'} = (f_{\mathcal{T}})_{\{1,\dots,n\}\setminus\{k\},x}$.

Proof. Take $\mathcal{T}'$ to be the composition $\mathcal{T}[k \leftarrow \mathcal{T}_x]$ where $\mathcal{T}_x$ is the two state transducer from Lemma 3.3.12 computing the finite number $x \in \mathrm{Fin}_{\mathcal{S}_k}$. The equation (3.6) is a special case of (3.3). □

3.4 States as functions

The main idea of this section is the following. During a computation of a function, a transducer moves from state to state while accepting input and producing output. In each state, one can abstract from the input and output made so far and imagine that the machine is starting afresh from this state. From this point of view, the transducer computes another function which is derived from the original function and from the input and output made so far.

Definition 3.4.1 (Changing the initial state).
Let $\mathcal{T} = (Q, \zeta, q_I)$ be a precise transducer. For every state $q \in Q$, define the transducer $\mathcal{T}^q := (Q, \zeta, q)$. The set and functions $\mathcal{H}_{\mathcal{T}^q}$, $\sigma_{\mathcal{T}^q}$, $\bar{f}_{\mathcal{T}^q}$ and $f_{\mathcal{T}^q}$ are said to be *computations of, respectively (completely) computed by,* $\mathcal{T}$ *from state* q. Let us put $\boldsymbol{\mathcal{H}^q_{\mathcal{T}}} := \mathcal{H}_{\mathcal{T}^q}$, $\boldsymbol{\sigma^q_{\mathcal{T}}} := \sigma_{\mathcal{T}^q}$, $\boldsymbol{\bar{f}^q_{\mathcal{T}}} := \bar{f}_{\mathcal{T}^q}$ and $\boldsymbol{f^q_{\mathcal{T}}} := f_{\mathcal{T}^q}$ for typographical reasons.

Remark 3.4.2. If $\mathcal{T}$ is total, precise or extensional, so is $\mathcal{T}^q$ for any *reachable* state q.

Proposition 3.4.3 (Formula for functions in reachable states).
Let $\mathcal{T} = (Q, \zeta, q_I)$ *be a precise IFS-transducer of type* $(\mathcal{S}_1, \ldots, \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$. *Let* $q \in Q$ *be a reachable state and* $\gamma \in \mathcal{H}^q_{\mathcal{T}}$ *be a finite computation of* $\mathcal{T}_q$. *Then it holds*

$$f^{q_\gamma}_{\mathcal{T}} \subseteq D_{\mathcal{S}_0}(\Pi_0(\gamma))^{-1} \circ f^q_{\mathcal{T}} \circ \big(D_{\mathcal{S}_1}(\Pi_1(\gamma)), \ldots, D_{\mathcal{S}_n}(\Pi_n(\gamma))\big). \tag{3.7}$$

(Recall that q_γ *is the final state of* γ *(page 37).) If* $\mathcal{T}$ *is extensional, the inclusion in (3.7) becomes equality.*

Proof. Suppose $y \in f^{q_\gamma}_{\mathcal{T}}(\vec{x})$. Take a computation $\gamma' \in \mathcal{H}^{q_\gamma}_{\mathcal{T}}$ with $[\![\Pi_i(\gamma')]\!] \ni x_i$ for all $i = 1, \ldots, n$, and $[\![\Pi_0(\gamma')]\!] = y$. Now $\gamma\gamma'$ is a computation from $\mathcal{H}^q_{\mathcal{T}}$ and it holds

$$[\![\Pi_i(\gamma\gamma')]\!] = D_{\mathcal{S}_i}\,(\Pi_i(\gamma))\,\big([\![\Pi_i(\gamma')]\!]\big) \ni D_{\mathcal{S}_i}\,(\Pi_i(\gamma))\,(x_i)$$

for all $i = 1, \ldots, n$, as well as $[\![\Pi_0(\gamma\gamma')]\!] = D_{\mathcal{S}_0}(\Pi_0(\gamma))(y)$. Therefore it holds

$$D_{\mathcal{S}_0}(\Pi_0(\gamma))(y) \in f^q_{\mathcal{T}}\Big(\big(D_{\mathcal{S}_1}(\Pi_1(\gamma))\big)(x_1), \ldots, \big(D_{\mathcal{S}_n}(\Pi_n(\gamma))\big)(x_n)\Big)$$

which is equivalent to

$$y \in \Big(D_{\mathcal{S}_0}(\Pi_0(\gamma))^{-1} \circ f^q_{\mathcal{T}} \circ \big(D_{\mathcal{S}_1}(\Pi_1(\gamma)), \ldots, D_{\mathcal{S}_n}(\Pi_n(\gamma))\big)\Big)\,(\vec{x}).$$

We finished the proof of (3.7). If $\mathcal{T}$ is extensional, $f^q_{\mathcal{T}}$ is single-valued and therefore the formula (3.7) holds as an equation because the left hand side is not partial. □

On the other hand, we will show that a transducer in which we can associate a function with each of its states so that (3.7) holds, really computes those functions (although not necessarily completely). The only extra condition for this to be true is that the transducer is precise. In case of finite transducers, this condition can be easily checked with the help of Lemma 3.3.3.

Proposition 3.4.4 (Constructing a transducer from functions).
Let $\mathcal{T} = (Q, \zeta, q_I)$ *be a precise finite transducer of type* $(\mathcal{S}_1, \ldots, \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ *and*

$$h\colon Q \to [I_{\mathcal{S}_1} \times \cdots \times I_{\mathcal{S}_n} \rightrightarrows I_{\mathcal{S}_0}]$$

an assignment of (many-valued total) continuous *functions to states such that for each* $q \in Q$ *and* $(q', u_1, \ldots, u_n, u_0) \in \zeta(q)$*, it holds*

$$h(q') \subseteq D_{\mathcal{S}_0}(u_0)^{-1} \circ h(q) \circ \big(D_{\mathcal{S}_1}(u_1), \ldots, D_{\mathcal{S}_n}(u_n)\big). \tag{3.8}$$

Then for each reachable state $q \in Q$ *it holds* $f^q_{\mathcal{T}} \subseteq h(q)$*, in particular,* $f_{\mathcal{T}} \subseteq h(q_I)$*, i.e.* $\mathcal{T}$ *computes* $h(q_I)$*.*

If all $h(q)$ *are single-valued, then* $\mathcal{T}$ *is extensional and it holds* $f^q_{\mathcal{T}} = h(q)$ *for all reachable* $q \in Q$*.*

(The conditions of reachability are necessary only because preciseness and extensionality are defined to cover only reachable states.)

Proof. Take arbitrary values $x_i \in I_{\mathcal{S}_i}$ and arbitrary representations $\alpha_i \in \Sigma^\omega_{\mathcal{S}_i}$ with $[\![\alpha_i]\!] = x_i$ for every $i = 1, \ldots, n$. Choose any computation $\gamma \in \mathcal{H}^q_{\mathcal{T}}(\alpha_1, \ldots, \alpha_n)$. Because $\mathcal{T}$ is total, γ is infinite and because $\mathcal{T}$ is talkative, $\Pi_0(\gamma)$ is also infinite.

Denote by $\gamma_0, \gamma_1, \gamma_2, \ldots \in \mathcal{H}^q_{\mathcal{T}}$ the ordered infinite sequence of all of the computations that are finite prefixes of γ: $\epsilon = \gamma_0 \sqsubseteq \gamma_1 \sqsubseteq \gamma_2 \sqsubseteq \cdots \sqsubseteq \gamma$. By successive application of (3.8) we get

$$h(q_{\gamma_k}) \subseteq D_{\mathcal{S}_0}(\Pi_0(\gamma_k))^{-1} \circ h(q) \circ \big(D_{\mathcal{S}_1}(\Pi_1(\gamma_k)), \ldots, D_{\mathcal{S}_n}(\Pi_n(\gamma_k))\big) \tag{3.9}$$

for all $k \in \mathbb{N}$. Denoting $I_{i,k} = [\![D_{\mathcal{S}_i}(\Pi_i(\gamma_k))]\!]$ for all $i = 0, \ldots, n$ and $k \in \mathbb{N}$, the fact that the right hand side of (3.9) is not partial, implies:

$$I_{0,k} \cap \big(h(q)\big)(x'_1, \ldots, x'_n) \neq \emptyset \text{ for all } x'_1 \in I_{1,k}, \ldots, x'_n \in I_{n,k} \tag{3.10}$$

It holds $x_i \in \bigcap_k I_{i,k}$ for every $i = 1, \ldots, n$ because $\Pi_i(\gamma) \sqsubseteq \alpha_i$. Since $\Pi_0(\gamma)$ is infinite, the intersection $\bigcap_k I_{0,k}$ is a singleton set, say $\{y\}$. Taking the limit case of (3.10) using continuity of $h(q)$, we get $y \in \big(h(q)\big)(x_1, \ldots, x_k)$. This is true for any $\gamma \in \mathcal{H}^q_{\mathcal{T}}(\alpha_1, \ldots, \alpha_n)$ and therefore for any $y \in f^q_{\mathcal{T}}(x_1, \ldots, x_n)$. Altogether we get $f^q_{\mathcal{T}}(x_1, \ldots, x_n) \subseteq \big(h(q)\big)(x_1, \ldots, x_k)$ for any $x_1, \ldots, x_n$, which is the first part of the proposition. The second part follows trivially from the first one. □

This proposition can be applied for constructing and checking finite transducers that compute certain real number functions. For example, in Figure 3.8, there are three simple unary finite transducers using the signed binary representation for both input and output. They are drawn via their underlying graphs. The initial state is marked by a bold arrow. Each state is associated with a function in such a way that the conditions of Proposition 3.4.4 are satisfied. This fact and Proposition 3.4.4 prove that the transducers really compute the functions that they are assigned with, i.e. the first one computes negation, the second $x/2$ and the third one a surjective function derived from the absolute value function. We have omitted the labels on the looping arrows on states B and C in the last transducer as they had already been described in the previous two transducers.

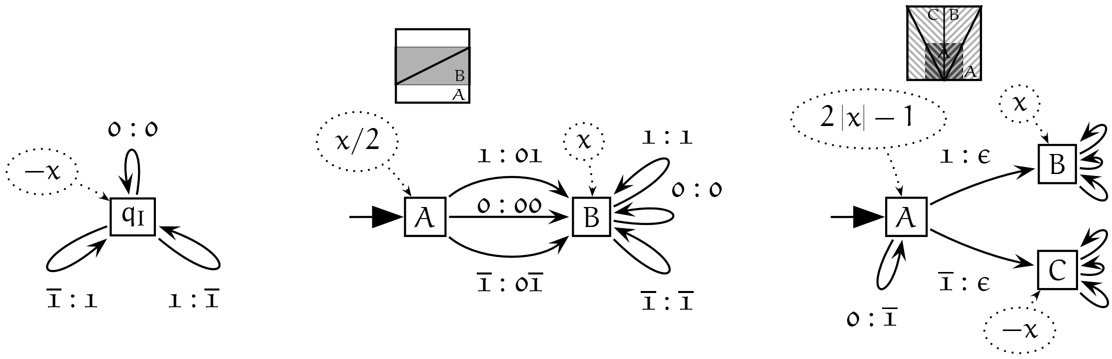

Figure 3.8: Examples of simple unary finite transducers with signed binary

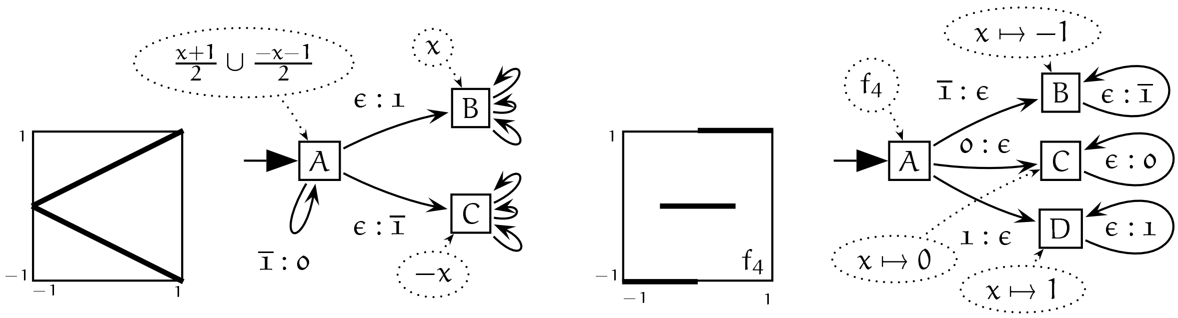

Figure 3.9: Examples of many-valued functions computed by finite transducers

The two transducers in Figure 3.9 completely compute truly many-valued functions whose graphs are illustrated to their left. The transducer on the left computes the inversion of $x \mapsto 2|x| - 1$ and can be obtained from the transducer computing $x \mapsto 2|x| - 1$ by swapping the input and output of each transition. Unfortunately, a transducer for inversion cannot be always obtained in this way because the 'inverted' transducer might not be even total. For example, notice that the loop on state A of the inverted transducer is superfluous. If we remove it, the transducer will still compute the inversion of $x \mapsto 2|x| - 1$ but its 'inverted' transducer would fail to be total and not compute $x \mapsto 2|x| - 1$.

The author believes that Proposition 3.4.4 can be applied to prove a result of theoretical interest which, together with an easy extension of Theorem 3.3.8, characterizes the many-valued functions expressible by a precise transducer (under the assumption that the output representation is open). Lisovik [1987] has formulated the single-valued analogy of such a characterization in the context of R-transducers. The additional conditions that one needs to impose on a *many-valued* function to guarantee that it can be completely computed by an (infinite non-recursive) transducer seem to be quite complicated yet interesting. They are a subject of current research.

Chapter 4

Finite Automata and Affine Representations

4.1 The limitation theorem

In this section, we will unfold in three steps the proof of the limitation theorem for transducers with affine representations, i.e. that every continuously differentiable thread of a finitely computable function of multiple arguments is affine. First we will consider strongly connected unary transducers then finite unary transducers and only then generalize to multiple input channel finite transducers. Recall that an oriented graph is strongly connected if for every two vertices, there is a path leading from one to the other.

4.1.1 Strongly connected unary transducers

Definition 4.1.1 (Strongly connected transducer).
A finite transducer $\mathcal{T}$ is *strongly connected (SC)* if its underlying graph $\mathcal{G}_{\mathcal{T}}$ is strongly connected.

SC transducers have the property that, to whichever state they might get during a computation, there is a possible continuation of this computation which would lead back to the initial state. This implies that through every state, there leads a cycle. Cycles in transducers are important for us because Proposition 3.4.3 has interesting implications for the case of a cycle. Formula (3.7) for a cycle in a unary transducer becomes a recursive inclusion of the following form:

Definition 4.1.2 (Sub-self-similarity).
Let $I_0, I_1 \in \mathcal{I}$, $g_i \in \mathrm{dContr}_{I_i}$ for $i = 0, 1$, and $f \colon I_1 \rightrightarrows I_0$ such that

$$f \subseteq g_0^{-1} \circ f \circ g_1. \tag{4.1}$$

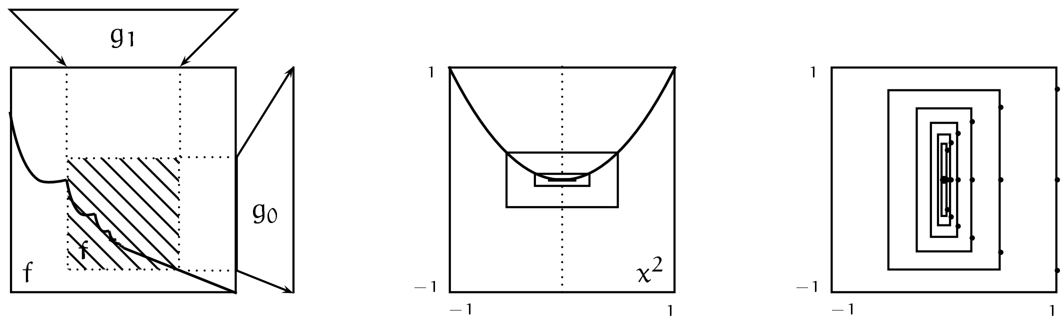

Figure 4.1: Affine sub-self-similarity of a function: balanced & mismatched strengths

This inclusion is called a *sub-self-similarity*[1] of f.

In the context of the above sub-self-similarity, the fixpoints Fix_{g_i} will be denoted by r_i and the strengths Strg_{g_i} by w_i for $i = 0, 1$. The fixpoint $(r_1, r_0) \in I_1 \times I_0$ of (g_1, g_0) is called the *centre* of the sub-self-similarity (4.1).

Sub-self-similarity is the key notion in our proof technique. Intuitively, it means that a rectangle section of the graph of a function is a transformed image of the whole graph of the function. When both d-contractions g_1 and g_0 are affine, the sub-self-similarity (4.1) can be visualized like in Figure 4.1 on the left. The crosshatched rectangle is given by the product of the domains of the d-contractions corresponding to input and output, respectively. When the graph of f is *clipped* by this rectangle and *enlarged* using the inverses of the d-contractions, we get a *superset* of the graph of f back again. (When f is single-valued, we get exactly the graph of f.)

This intuition is confirmed by the following simple and useful lemma.

Lemma 4.1.3 (Graph similarity).
In the situation of Definition 4.1.2, if $(x, y) \in \mathrm{Gr}(f)$ *then also* $(g_1(x), g_0(y)) \in \mathrm{Gr}(f)$. *Moreover, if* f *is continuous or single-valued in* r_1, *it holds* $(r_1, r_0) \in \mathrm{Gr}(f)$.

Proof. Since $y \in f(x)$, by formula (4.1) also $y \in g_0^{-1}(f(g_1(x)))$ which is equivalent to $g_0(y) \in f(g_1(x))$.

Applying the value r_1 to the functions in formula (4.1), we get $f(r_1) \subseteq g_0^{-1}(f(r_1))$ which is equivalent to $g_0(f(r_1)) \subseteq f(r_1)$. Now for any $z \in f(r_1)$, it holds $g_0^k(z) \in f(r_1)$ for all $k \in \mathbb{N}$. If f is single-valued in r_1, it holds $g_0(z) = z$, i.e. $f(r_1) = z = r_0$. If f is continuous, the limit of $g_0^k(x)$, equal to r_0 by Lemma 3.1.3, is in $f(r_1)$. □

A sub-self-similarity translates a lot of useful properties which a function has near the centre to its whole graph. For example, a thread is a locally single-valued piece of a many-valued func-

[1] We do not call it self-similarity (even in the case of a single-valued function when the inclusion becomes an equality) to avoid confusion with related but different self-similarity in fractal theory.

tion. It follows that when the centre of a sub-self-similarity is on a thread, the whole function has to be single valued.

Lemma 4.1.4 (Thread in the centre of a sub-self-similarity).
In the situation of Definition 4.1.2, if f *has a thread* $h\colon J \to I_0$ *with* $r_1 \in J$ *and* $h(r_1) = r_0$*, then* f *is single-valued.*

Proof. Let $y_1, y_2 \in f(x)$ for an arbitrary $x \in I_1$. By Lemma 4.1.3, we deduce that $g_0^k(y_1), g_0^k(y_2) \in f(g_1^k(x))$ for any $k \in \mathbb{N}$. Since $\mathrm{Zone}(h)$ is a neighbourhood of (r_1, r_0) within $I_1 \times I_0$, there is a $k \in \mathbb{N}$ for which

$$\left(g_1^k(x), g_0^k(y_1)\right), \left(g_1^k(x), g_0^k(y_2)\right) \in \mathrm{Zone}(h).$$

Thus $g_0^k(y_1) = g_0^k(y_2)$ by the definition of $\mathrm{Zone}(h)$. Now consider that g_0^k, as a d-contraction, is injective. This implies $y_1 = y_2$. Thus f is single-valued on the whole of I_1. □

We will show several other important consequences of sub-self-similarity. The first one is concerned with the ratio of convergence speeds of input and output. At this point we restrict ourselves to *affine* d-contractions only. The version for general d-contractions will follow in Chapter 5 once we have a general version of Lemma 3.1.6 on page 27 available. The main ideas of the proof are visualized in Figure 4.1 in the middle and on the right hand side.

Lemma 4.1.5 (Strength mismatch in affine sub-self-similarity).
In the situation of Definition 4.1.2, assuming also that $(r_1, r_0) \in \mathrm{Gr}\,(f)$ *and that both of the contractions* g_i *are affine, it holds:*

1a. If $w_1 > w_0$ *(input is weaker than output), then the many-valued weak derivative* $f^{\mathrm{wk}}_{(r_0)}(r_1)$ *contains* 0*.*

1b. If, moreover, f *is single-valued, the ordinary derivative* $f'(r_1)$ *exists and is equal to* 0*.*

2a. If $w_1 < w_0$ *(input is stronger than output), then the many-valued weak derivative* $f^{\mathrm{wk}}_{(r_0)}(r_1)$ *contains* 0 *or* ∞*.*

2b. If, moreover, f *is single-valued, continuous and injective on some neighbourhood of* r_1*, the ordinary derivative* $f'(r_1)$ *exists and is equal to* $+\infty$ *or* $-\infty$*.*

Proof. From equation (4.1) we get $f \subseteq g_0^{-k} \circ f \circ g_1^k$ for any natural k:

$$f \subseteq g_0^{-1} \circ f \circ g_1 \subseteq g_0^{-2} \circ f \circ g_1^2 \subseteq \cdots$$

Supposing that g_0 and g_1 are affine, by Lemma 3.1.6, it holds $\left|[\![g_i^k]\!]\right| = |I_i| \cdot w_i^k$ for every k. Similarly, one can show that

$$\min\left\{|x - r_i| \;\middle|\; x \in [\![g_i^k]\!] \setminus [\![g_i^{k+1}]\!]\right\} = \min\left\{|x - r_i| \;\middle|\; x \in I_i \setminus [\![g_i]\!]\right\} \cdot w_i^k. \tag{4.2}$$

Take any $x \in I_1 \setminus \{r_1\}$ and $y \in f(x)$. By Lemma 4.1.3, $g_0^k(y) \in f(g_1^k(x))$ holds for any $k \in \mathbb{N}$. Now we can calculate a particular value of the weak derivative $f^{wk}_{(r_0)}(r_1)$ as follows:

$$\lim_{k\to\infty} \frac{g_0^k(y) - r_0}{g_1^k(x) - r_1} = \lim_{k\to\infty} \frac{(y - r_0) \cdot w_0^k}{(x - r_1) \cdot w_1^k} \tag{4.3}$$

which is equal to 0 if $w_0 < w_1$. If $w_0 > w_1$ then the limit is equal either to ∞, when $y \neq r_0$, or to 0, when $y = r_0$. Thus the claims 1a and 2a have been proved.

Assuming that f is single-valued, we will aim at proving the existence of the derivative $f'(r_1)$.

In order to prove 1b, assume $w_0 < w_1$. Take an *arbitrary* sequence $x_1, x_2, \dots$ of points in $I_1 \setminus \{r_1\}$ which converges to r_1. For each x_j let k_j be the only exponent for which $x_j \in [\![g_1^{k_j}]\!]$ and at the same time $x_j \notin [\![g_1^{k_j+1}]\!]$. Notice that $\lim_{j\to\infty} k_j = \infty$ and that $f(x_j) \in \left[\!\!\left[g_0^{k_j}\right]\!\!\right]$ for all j. Now we can calculate:

$$\begin{aligned}\lim_{j\to\infty} \frac{|f(x_j) - r_0|}{|x_j - r_1|} \leqslant \lim_{j\to\infty} & \frac{\left|[\![g_0^{k_j}]\!]\right|}{\min\{|x - r_1| \mid x \in [\![g_1^{k_j}]\!] \setminus [\![g_1^{k_j+1}]\!]\}} = \\ & \lim_{j\to\infty} \frac{|I_0| \cdot w_0^{k_j}}{\min\{|x - r_1| \mid x \in I_1 \setminus [\![g_1]\!]\} \cdot w_1^{k_j}} = D \cdot \lim_{j\to\infty} \left(\frac{w_0}{w_1}\right)^{k_j} = 0 \end{aligned} \tag{4.4}$$

where D is a constant. This proves that the derivative $f'(r_1)$ exists and is equal to 0.

In order to prove 2b, assume $w_0 > w_1$ and at the same time let f be injective on a neighbourhood U of r_1. Take an arbitrary sequence $x_1, x_2, \dots$ of points in $U \setminus \{r_1\}$ converging to r_1. It holds $f(x_j) \neq r_0$ for all $j \in \mathbb{N}$ because f is injective on U. This allows us to define k_j for each x_j to be the exponent for which $f(x_j) \in [\![g_0^{k_j}]\!]$ and at the same time $f(x_j) \notin [\![g_0^{k_j+1}]\!]$. Again, it holds $\lim_{j\to\infty} k_j = \infty$ and $x_j \in \left[\!\!\left[g_1^{k_j}\right]\!\!\right]$, and we can calculate:

$$\begin{aligned}\lim_{j\to\infty} \frac{|f(x_j) - r_0|}{|x_j - r_1|} \geqslant \lim_{j\to\infty} & \frac{\min\{|x - r_0| \mid x \in [\![g_0^{k_j}]\!] \setminus [\![g_0^{k_j+1}]\!]\}}{\left|[\![g_1^{k_j}]\!]\right|} = \\ & \lim_{j\to\infty} \frac{\min\{|x - r_0| \mid x \in I_0 \setminus [\![g_0]\!]\} \cdot w_0^{k_j}}{|I_1| \cdot w_1^{k_j}} = E \cdot \lim_{j\to\infty} \left(\frac{w_0}{w_1}\right)^{k_j} = \infty \end{aligned} \tag{4.5}$$

where E is a constant. Since f is continuous and injective, the sign of $(f(x_j) - r_0)/(x_j - r_1)$ is constant. This and (4.5) prove that the derivative $f'(r_1)$ exists and is infinite, i.e. equal to $+\infty$ or $-\infty$. □

The single-valued corollary of the following lemma allows us to extend the local property of 'having a non-zero derivative in the centre of a sub-self-similarity' to a global property of 'being affine'.

Lemma 4.1.6 (Affine sub-self-similarity determines the function).
In the situation of Definition 4.1.2, if both of the contractions g_i are affine, the sub-self-similarity (4.1) implies that:

1. *either* $f^{wk}_{(r_0)}(r_1) \cap \{0, \infty\} \neq \emptyset$
2. *or* $f(x) \subseteq f^{wk}_{(r_0)}(r_1) \cdot (x - r_1) + r_0$.

Corollary 4.1.7. *If* f *is single-valued, in the preceding lemma, one can replace 1 and 2 with the following:*

1. *either* $f'(r_1) \in \{0, +\infty, -\infty\}$
2. *or* $f'(r_1)$ *doesn't exist*
3. *or* $f'(r_1)$ *exists and* f *is affine:* $f(x) = f'(r_1) \cdot (x - r_1) + r_0$.

Proof of Lemma 4.1.6. If $w_1 \neq w_0$, we can apply Lemma 4.1.5 to conclude 1. Assume that $w_1 = w_0$ for the rest of the proof.

Since g_i are affine, it holds $g_i^k(x) - r_i = (x - r_i) \cdot w_i^k$ for every $x \in I_i$ and $k \in \mathbb{N}$ by Lemma 3.1.6.

Take an arbitrary point $x \in I_1$, $x \neq r_1$, and pick $y \in f(x)$. Define $x_k := g_1^k(x)$ and $y_k := g_0^k(y)$ for all $k \in \mathbb{N}$. Now a value of the derivative $f^{wk}_{(r_0)}(r_1)$ can be expressed by

$$f^{wk}_{(r_0)}(r_1) \ni \lim_{k\to\infty} \frac{y_k - r_0}{x_k - r_1} = \lim_{k\to\infty} \frac{(y - r_0) \cdot w_0^k}{(x - r_1) \cdot w_1^k} = \frac{y - r_0}{x - r_1}$$

out of which the desired formula for y can be extracted: $y \in f^{wk}_{(r_0)}(r_1)\,(x - r_1) + r_0$. □

Now we have established enough knowledge about sub-self-similarity to be able to prove the first, weaker version of the main limitation theorem.

Proposition 4.1.8 (The main limitation theorem for SC unary affine transducers).
Let $\mathcal{T} = (Q, \zeta, q_I)$ *be a precise non-ignorant strongly connected IFS-transducer of type* $(\mathcal{S}_1) \rightrightarrows \mathcal{S}_0$ *with both representations* $\mathcal{S}_i$ *being affine. Then one of the following holds:*

1. $f_{\mathcal{T}}$ *does not have any threads.*
2. $f_{\mathcal{T}}$ *is single-valued and there is a set* E *dense in* $I_{\mathcal{S}_1}$*, such that one of the following holds:*
   a) for each $x \in E$, $f'_{\mathcal{T}}(x)$ *is undefined.*
   b) there exists $d \in \{0, +\infty, -\infty\}$ *such that for each* $x \in E$, $f'_{\mathcal{T}}(x) = d$.
3. $f_{\mathcal{T}}$ *is single-valued and affine on the whole of* $I_{\mathcal{S}_1}$.

Proof. Assume now that 1 is not true and aim at proving 2 or 3. Take a thread $h\colon J \to I_{\mathcal{S}_0}$ of $f_{\mathcal{T}}$ and define

$$C_h := \left\{ \gamma \in \mathcal{H}^{\text{fin}}_{\mathcal{T}} \,\middle|\, q_\gamma = q_I,\ [\![\Pi_1(\gamma)]\!] \times [\![\Pi_0(\gamma)]\!] \subseteq \text{Zone}(h) \right\} \quad \text{and} \quad E_h := \{r_\gamma \mid \gamma \in C_h\}$$

where $r_\gamma := \mathrm{Fix}_{D_{\mathcal{S}_1}(\Pi_1(\gamma))}$ for each $\gamma \in \mathcal{H}_{\mathcal{T}}^{\mathrm{fin}}$. Notice that the properties of Zone(h) imply that for each $\gamma \in C_h$, it holds

$$[\![\Pi_0(\gamma)]\!] \cap f(r_\gamma) = \{h(r_\gamma)\}$$

This means that C_h is the set of the computations that will lead the transducer $\mathcal{T}$ back to the initial state and at the same time, if iterated forever, would compute one of the values of the thread h.

We will now show that E_h is dense in J. Take any open $U \subseteq J$ and any $x \in U$. Since $h(x) \in f_{\mathcal{T}}(x)$, there is a computation $\gamma \in \mathcal{H}_{\mathcal{T}}$ with $[\![\Pi_1(\gamma)]\!] = x$ and $[\![\Pi_0(\gamma)]\!] = h(x)$. Since Zone(h) is open, one can find an open rectangle $J_1 \times J_0$ with $x \in J_1 \subseteq U$, $h(x) \in J_0$ and $J_1 \times J_2 \subseteq \mathrm{Zone}(h)$. By Lemma 3.2.8 and the assumption that $\mathcal{T}$ is not ignorant and thus $\Pi_1(\gamma)$ is infinite, there is a prefix γ_1 of γ with $[\![\Pi_1(\gamma_1)]\!] \times [\![\Pi_0(\gamma_1)]\!] \subseteq J_1 \times J_2$.

Since $\mathcal{T}$ is SC, there is a continuation γ_2 of the computation γ_1 such that $\gamma_1\gamma_2 \in C_h$. Therefore there is a number in E_h, namely $r_{\gamma_1\gamma_2}$, which is in U. This concludes the proof that E_h is dense in J and, in particular, not empty.

For each $\gamma \in C_h$, by applying equation (3.7) from Proposition 3.4.3 on page 46, we derive the sub-self-similarity

$$f_{\mathcal{T}} \subseteq D_{\mathcal{S}_0}(\Pi_0(\gamma))^{-1} \circ f_{\mathcal{T}} \circ D_{\mathcal{S}_1}(\Pi_1(\gamma)). \tag{4.6}$$

There exists such a sub-self-similarity because C_h is not empty. Since $[\![\Pi_1(\gamma)]\!] \times [\![\Pi_0(\gamma)]\!] \subseteq \mathrm{Zone}(h)$, it holds $h(r_\gamma) = \mathrm{Fix}_{D_{\mathcal{S}_0}(\Pi_0(\gamma))}$. Therefore by Lemma 4.1.4, we get that $f_{\mathcal{T}}$ is single-valued. Because $f_{\mathcal{T}}$ is also continuous by Theorem 3.3.8, $f_{\mathcal{T}}$ is a thread of itself and we can put $h := f_{\mathcal{T}}$.

Assuming that 3 does not hold, i.e. that $f_{\mathcal{T}}$ is not affine, we will prove 2. Pick some $\widehat{\gamma} \in C_h$. By Corollary 4.1.7, we get that $f'_{\mathcal{T}}(r_{\widehat{\gamma}})$ either does not exist or is equal to 0, $+\infty$ or $-\infty$. Put

$$E := \left\{ \left(D_{\mathcal{S}_1}(\Pi_1(\gamma))\right)(r_{\widehat{\gamma}}) \;\middle|\; \gamma \in C_h \right\}.$$

By the sub-self-similarity (4.6), we can deduce that for each $\gamma \in C_h$, $f'_{\mathcal{T}}\big(\left(D_{\mathcal{S}_1}(\Pi_1(\gamma))\right)(r_{\widehat{\gamma}})\big)$ exists if and only if $f'_{\mathcal{T}}(r_{\widehat{\gamma}})$ exists and if they exists, they are equal (due to the fact that $f'_{\mathcal{T}}(r_{\widehat{\gamma}})$ is restricted to $\{0, +\infty, -\infty\}$). This follows from the chain rule and the fact that all d-contractions are differentiable with derivative not equal to 0, $+\infty$ nor $-\infty$ anywhere.

All we need to prove now is that E is dense in $I_{\mathcal{S}_1}$. This can be proved in exactly the same way like the density of E_h because the only property of E_h that the proof used was that it is a set of points chosen from the intervals $[\![\Pi_1(\gamma)]\!]$ for $\gamma \in C_h$, one point per interval. The set E is of the same form. □

4.1.2 Unary finite transducers

How to extend Proposition 4.1.8 to transducers that are not SC? We will see that each transducer $\mathcal{T}$ has at least one state q such that $\mathcal{T}^q$ is SC. Thus in any transducer, there are some states to which Proposition 4.1.8 applies. In the following, we will prepare a method for proving that a property holds for each vertex of an oriented graph (i.e. for every state of a transducer, in our case) by induction. This will allow us to extend an essential part of Proposition 4.1.8 to arbitrary states of finite transducers. These ideas are based on elementary graph theory and do not contain any new insight in themselves. We state them here only as a preparation for the proof of the limitation theorem.

Definition 4.1.9 (Strongly connected components).
Let $\mathcal{G}$ be an oriented graph. Denote the set of its vertices by $V_{\mathcal{G}}$. The expression $q \rightarrow_{\mathcal{G}} q'$ for vertices $q, q' \in V_{\mathcal{G}}$ means that there is an edge leading from q to q' in $\mathcal{G}$, while another expression, $q \rightarrow^*_{\mathcal{G}} q'$, means that there is a path from q to q' in $\mathcal{G}$.

Using the reflexive and transitive relation $\rightarrow^*_{\mathcal{G}}$, one defines the following equivalence relation on vertices:

$$q \leftrightarrows^*_{\mathcal{G}} q' \text{ if and only if both } q \rightarrow^*_{\mathcal{G}} q' \text{ and } q' \rightarrow^*_{\mathcal{G}} q$$

which is pronounced: 'q *is strongly connected with* q''. The equivalence classes of $\leftrightarrows^*_{\mathcal{G}}$ are called the *strongly connected components* of $\mathcal{G}$. The $\leftrightarrows^*_{\mathcal{G}}$ class that contains $q \in V_{\mathcal{G}}$ we will denote by $[q]_{\leftrightarrows^*_{\mathcal{G}}}$.

Whenever there is no confusion about which graph they relate to, the subscript $\mathcal{G}$ can be left out from the symbols $V_{\mathcal{G}}$, $\rightarrow_{\mathcal{G}}$, $\rightarrow^*_{\mathcal{G}}$, $\leftrightarrows^*_{\mathcal{G}}$ as well as from other symbols relative to $\mathcal{G}$ defined later.

When factored by the equivalence of being strongly connected, a graph becomes acyclic. Therefore it is possible to define the depth of a vertex as the distance from the associated root as follows:

Definition 4.1.10 (Depth of a vertex).
For a vertex q of a finite oriented graph $\mathcal{G}$, let $depth_{\mathcal{G}}(q)$ denote the maximal number of $\leftrightarrows^*$ equivalence classes that the vertices on a path leading from q may belong to:

$$\boldsymbol{depth}_{\mathcal{G}}(q) := \\ \max\left\{\text{card}\left\{[q_i]_{\leftrightarrows^*} \,\middle|\, i = 1, \dots, k\right\} \,\middle|\, k > 1,\ q_1, \dots, q_k \in V,\ q = q_1, q_1 \rightarrow \dots \rightarrow q_k\right\}$$

where max over the empty set is 0. For each $n \in \mathbb{N}$, define

$$SC^n_{\mathcal{G}} = \left\{q \in V \,\middle|\, depth(q) \leqslant n\right\}$$

.

Remark 4.1.11 (Basic properties of depth).
In any finite oriented graph

1. the set SC^0 is equal to the set of vertices without any outgoing edge,
2. it holds: $SC^0 \subseteq SC^1 \subseteq SC^2 \subseteq \dots,$
3. for any pair of vertices q, q', it holds:
   $q \to^* q' \Longrightarrow (q \leftrightarrows^* q' \text{ or } \mathit{depth}(q) > \mathit{depth}(q'))$.
4. For every vertex q of a finite oriented graph it holds:
   $\mathit{depth}(q) = n > 1 \Longrightarrow (\exists q')\big(q \to^* q' \text{ and } \mathit{depth}(q') < n\big)$

Lemma 4.1.12. *Let $\mathcal{G}$ be a nonempty finite oriented graph in which each vertex has at least one outgoing edge. Then it holds $SC^0 = \emptyset$ and $SC^1 \neq \emptyset$.*

Proof. $SC^0 = \emptyset$ follows from the previous remark, point (1). $SC^1 \neq \emptyset$ then follows from point (4) of the previous remark as follows: Let m be the smallest natural number for which $SC^m \neq \emptyset$. Such an m is well defined because $\mathcal{G}$ is finite and non-empty and therefore there exists a non-empty SC^k for some k. By point (4), this m has to be equal to 1 because otherwise we would find a vertex with depth $j < m$ which would mean $SC^j \neq \emptyset$, a contradiction to the minimality of m. □

Now we have all ingredients ready for proving the unary version of the main limitation theorem by induction on the depth of vertices in the underlying graph.

Proposition 4.1.13 (Unary Affine version of the Main Limitation Theorem).
Let $\mathcal{T}$ be a precise non-ignorant finite transducer of type $(\mathcal{S}_1) \rightrightarrows \mathcal{S}_0$ with both representations $\mathcal{S}_i$ being affine. Then all continuously differentiable threads of the function $f_\mathcal{T}$ are affine.

Proof. We prove the proposition for every $f^q_\mathcal{T}$ instead of $f_\mathcal{T}$ only. We do it by induction on the depth of q in the underlying graph $\mathcal{G}_\mathcal{T}$. Notice that the case of $\mathit{depth}(q) = 1$ does not need to be considered separately. We do so only in order to make the proof easier to understand.

For $\mathit{depth}(q) = 1$, the sub-transducer consisting of the states reachable from q is SC. We can apply Proposition 4.1.8 to get that either $f^q_\mathcal{T}$ does not contain any thread or it is single-valued and either affine on the whole of $I_{\mathcal{S}_1}$ or lacking derivative or having derivative equal to d for some $d \in \{0, +\infty, -\infty\}$, on a dense set E. Assume, for the sake of contradiction, that a thread $h\colon J \to I_0$ of $f^q_\mathcal{T}$ is continuously differentiable but not affine. Then the derivative of h has to be equal to one of the three values above on the set $E \cap J$, which is dense in J. By the continuity of h', we get that h' is constant. Since a derivative cannot be constantly infinite, it has to be constantly 0. Thus h is constant, i.e. affine, which is a contradiction.

Assume, as the induction hypothesis, that the proposition holds for all $f^{q'}_{\mathcal{J}}$ for vertices q' with *depth*$(q') < k$, i.e. that all threads of $f^{q'}_{\mathcal{J}}$ are either affine or not continuously differentiable. Take a state q with *depth*$(q) = k$ and a thread $h\colon J \to I_{\mathcal{S}_0}$ of $f^{q}_{\mathcal{J}}$.

Let

$$H := \left\{\gamma \in \mathcal{H}^{\text{fin}}_{\mathcal{J}} \,\middle|\, [\![\Pi_1(\gamma)]\!] \times [\![\Pi_0(\gamma)]\!] \subseteq \text{Zone}(h)\right\} \quad \text{and} \quad F := \{r_\gamma \,|\, \gamma \in H\}$$

where $r_\gamma := \text{Fix}_{D_{\mathcal{S}_1}(\Pi_1(\gamma))}$. These are: a set of finite computations whose arbitrary continuation leads to a point of the thread h, and the set of fixpoints of their inputs. From Lemma 3.2.8, it follows that F is dense in J. If for all $x \in F$, it holds $h'(x) \in \{0, +\infty, -\infty\}$, then we can conclude (similarly like in the proof of Proposition 4.1.8 above, with the assumption that h is continuously differentiable) that the derivative h' is constantly 0 and thus h is constant on J. Assume, therefore, that there is a $\gamma_1 \in H$ for which $h'(r_{\gamma_1}) \notin \{0, +\infty, -\infty\}$.

It follows from Remark 4.1.11, point (4) that for γ_1, as well as for any other computation, it holds either *depth*$(q_{\gamma_1}) < k$ or there is a γ_2 such that $q_{\gamma_1\gamma_2} = q$.

In the latter case, we have a sub-self-similarity whose centre has to be on the thread h. Therefore, the weak derivative of f in the centre of this sub-self-similarity has exactly one value (i.e. the derivative of h), and one can apply Lemma 4.1.6 to conclude that $f^{q}_{\mathcal{J}}$ is single-valued and affine on the whole of $I_{\mathcal{S}_1}$.

In the former case, we use the induction hypothesis for q_{γ_1}. Let us apply formula (3.7) in Proposition 3.4.3 on page 46 to γ_1:

$$f^{q_{\gamma_1}}_{\mathcal{J}} \subseteq D_{\mathcal{S}_0}(\Pi_0(\gamma_1))^{-1} \circ f^{q}_{\mathcal{J}} \circ \left(D_{\mathcal{S}_1}(\Pi_1(\gamma_1)), \ldots, D_{\mathcal{S}_n}(\Pi_n(\gamma_1))\right).$$

From it, we can deduce that the function $f^{q_{\gamma_1}}_{\mathcal{J}}$ is dependent on h and γ_1 only and thus it is single-valued and continuously differentiable on the whole of $I_{\mathcal{S}_1}$. Therefore, by the induction hypothesis, $f^{q_{\gamma_1}}_{\mathcal{J}}$ has to be affine. Again, by the formula above, we get that h is affine on the non-singleton interval $[\![\Pi_1(\gamma_1)]\!]$. Let $[a, b] := [\![\Pi_1(\gamma_1)]\!]$.

This situation permits us to define

$$z := \min\left\{x \in \overline{J} \,\middle|\, h|_{[x,b]} \text{ is affine}\right\} \text{ and } Z := \max\left\{x \in \overline{J} \,\middle|\, h|_{[a,x]} \text{ is affine}\right\}$$

where $\overline{J}$ is the closure of J. This means that $[z, Z]$ is a maximal interval on which h is affine. Since $[a, b] \subseteq [z, Z]$, the interval $[z, Z]$ is not singleton. Recall that $h|_{[z,Z]}$ is not constant because the derivative $h'(r_{\gamma_1})$ is assumed to be non-zero. We will finish the proof by showing that $[z, Z] = \overline{J}$. This will follow from $z, Z \notin J$ because J is an open interval.

In order to pursue a contradiction, suppose that $z \in J$. Take a representation $\alpha \in \Sigma^{\omega}_{\mathcal{S}_1}$ of z such that $[\![\alpha[1, j]]\!] \setminus [z, Z]^o \neq \emptyset$ for each j which is possible by Lemma 3.2.11 on page 33. Since J is open, there is an index j_0 such that $[\![\alpha[1, j]]\!] \subseteq J$ for all $j \geqslant j_0$. For each $j \geqslant j_0$, either h is affine on $[\![\alpha[1, k]]\!]$ or there is a point $x_j \in [\![\alpha[1, j]]\!]$ with $h'(x_j) \in \{0, +\infty, -\infty\}$. In the former case, h

would be affine on $[z, Z] \cup [\![\alpha[1, j]]\!]$ which is larger than $[z, Z]$ and thus contradicts the choice of z. If it is the latter case for all $j > j_0$, the sequence x_j converges to z and by continuity of the derivative of h on J, its derivative in z would have to be 0, $+\infty$ or $-\infty$, which contradicts the fact that $f^q_{\mathcal{J}}$ is affine and non-constant on $[z, Z]$.

The proof of $Z \notin J$ is symmetrical. $\square$

The author believes that this proposition could be strengthened by replacing the condition of continuous differentiability with differentiability, using the technique of Lisovik & Shkaravskaya [1998]. The statement would not hold without the condition of differentiability or a similar smoothness condition even using an open representation like signed binary. This can be observed, for example, on the SC finite transducer in Example 4.2.8 on page 72 which computes a function which is not continuously differentiable on any subinterval of the domain $[-1, 1]$.

4.1.3 Multiple input finite transducers

In this final subsection, we are going to extend Proposition 4.1.13 to functions with multiple arguments.

Unfortunately, the multi-dimensional affine version of the limitation theorem does not follow from Proposition 4.1.13 applied at all unary instantiations of a function because there are functions that are not affine although all of its unary instantiations are. For example, the binary function $(x, y) \mapsto x \cdot y$ is not affine but for each $t \in \mathbb{R}$, the functions $x \mapsto x \cdot t$ and $y \mapsto t \cdot y$ are affine. For this reason, we need to go back to sub-self-similarity and extend it to multiple dimensions.

Definition 4.1.14 (Multi-dimensional sub-self-similarity).
Let $I_0, I_1, \ldots, I_n \in \mathcal{I}$, $g_i \in \mathrm{dContr}_{I_i}$ for $i = 0, \ldots, n$, and $f\colon I_1 \times \cdots \times I_n \rightrightarrows I_0$ be such a function that the following inclusion holds:

$$f \subseteq g_0^{-1} \circ f \circ (g_1, \ldots, g_n) \tag{4.7}$$

This formula is called an n-*ary sub-self-similarity* of f. The fixpoints Fix_{g_i} for $i = 0, \ldots, n$, in the context of the above sub-self-similarity, will be denoted by r_i and the strengths Strg_{g_i} by w_i. Denote also $\vec{r} := (r_1, \ldots, r_n)$, $\vec{g} = (g_1, \ldots, g_n)$ and $V = I_1 \times \ldots \times I_n$ as usual. The point $(\vec{r}, r_0) \in V \times I_0$ is called the *centre* of the sub-self-similarity (4.7).

For multi-dimensional sub-self-similarity, the same intuition applies like for the unary one. Therefore we get some analogies of the basic properties from the unary case (see below).

The proofs of the following two lemmas are exactly the same like the ones for the unary variants except for replacing x with $\vec{x}$, g_1 with $\vec{g}$ and I_1 with V.

Lemma 4.1.15 (Graph similarity).
If f, $g_1, \dots, g_n$ *satisfy (4.7) then*

$$(\vec{x}, y) \in \mathrm{Gr}\,(f) \Longrightarrow (\vec{g}(\vec{x}), g_0(y)) \in \mathrm{Gr}\,(f)\,.$$

Moreover, if f *is continuous or single-valued in* $\vec{r}$*, it holds* $r_0 \in f(\vec{r})$.

Lemma 4.1.16 (Thread in the centre of a multi-dimensional sub-self-similarity).
In the situation of Definition 4.1.14, if f *has a thread* $h\colon J \to I_0$ *and* $h(\vec{r}) = r_0$*, then* f *is single-valued.*

Next we will formulate and prove an analogy of Lemma 4.1.6.

There is a new difficulty in formulating this result when there are multiple arguments. In each coordinate independently, the partial derivative in the centre of a sub-self-similarity can contain one of the values 0, $+\infty$, $-\infty$ or not. Therefore we need to consider a set of coordinates in which this does not happen and then we will be able to say something about the instantiation to these coordinates.

The statement below says much more about a function with a sub-self-similarity than just that each unary instantiation fulfils Lemma 4.1.6. It can say something globally about an instantiation to *all* those coordinates in which the condition on the partial derivative holds.

Lemma 4.1.17 (Affine multi-dimensional sub-self-similarity determines the function).
In the situation of Definition 4.1.14, if all of the contractions g_i *are affine, the sub-self-similarity (4.7) implies the following statements:*

If for some $P \subseteq \{1, \dots, n\}$*, the instantiation* $\widehat{f} := f_{P,\vec{r}_{\overline{P}}}$ *is differentiable in* $\vec{r}_P$ *and for every* $k \in P$*, the partial derivative* $\left(f_{k,(r_i)_{i \neq k}}\right)'_{(r_0)}(r_k)$ *does not include zero, then it holds:*

$$\widehat{f}(\vec{x}) \subseteq \left(\mathbf{d}\widehat{f}_{(r_0)}(\vec{r}_P)\right)(\vec{x} - \vec{r}_P) + r_0 \quad \textit{for all } \vec{x} \in \textstyle\prod_{i \in P} I_i$$

If f *is single-valued, then the above implies that* $\widehat{f}$ *is affine on* $\prod_{i \in P} I_i$.

Proof. From (4.7), a sub-self-similarity for $\widehat{f}$ can be derived:

$$\widehat{f} = f_{P,\vec{r}_{\overline{P}}} \subseteq \left(g_0^{-1} \circ f \circ \vec{g}\right)_{P,\vec{r}_{\overline{P}}} = g_0^{-1} \circ f_{P,\vec{r}_{\overline{P}}} \circ \vec{g}_P = g_0^{-1} \circ \widehat{f} \circ \vec{g}_P.$$

Therefore it is sufficient to prove the lemma for $P = \{1, \dots, n\}$ and $\widehat{f} = f$ since for a smaller P we may completely ignore the arguments which are not in P. Let us now assume that f is differentiable in $\vec{r} = (r_1, \dots, r_n)$ and that all partial derivatives $\left(f_{k,(r_i)_{i \neq k}}\right)'_{(r_0)}(r_k)$ in that point do not include zero.

Next we will prove that $w_0 = w_1 = \cdots = w_n$. Suppose, on the contrary, that $w_0 \neq w_k$ for some $k \in \{1, \dots, n\}$. The unary instantiation $f_k := f_{k,(r_i)_{i \neq k}}$ of f satisfies a (unary) sub-self-similarity

$$f_k \subseteq g_0^{-1} \circ f_k \circ g_k.$$

Applying Lemma 4.1.5 to f_k we get that $(f_k)'_{(r_0)}(r_k)$ contains either 0 or ∞. Since $(f_k)'_{(r_0)}(r_k)$ is equal to the partial derivative $\left(f_{j,(r_i)_{i\neq j}}\right)'_{(r_0)}(r_j)$, it cannot contain 0. It cannot contain infinity either because the definition of differentiability in higher dimensions excludes it—a contradiction. Since we have proved that $w_0, w_1, \dots, w_n$ are all equal, denote their common value simply by w.

Pick an arbitrary point $(\vec{x}, y) \in \mathrm{Gr}(f)$ and define the sequences

$$\vec{x}^{(j)} := \vec{g}^{j}(\vec{x}) \quad \text{and} \quad y^{(j)} := g_0^{j}(y) \quad \text{for all } j \in \mathbb{N}$$

which converge to $\vec{r}$ and r_0, respectively. By Lemma 4.1.15, $y^{(j)} \in f(\vec{x}^{(j)})$ for all j.

Using the definition of many-valued differentiability (Definition 2.6.4 on page 22) and the fact that all $g_0, \dots, g_n$ are affine with the same strength, we get the following formula:

$$\frac{y - r_0}{|\vec{x} - \vec{r}|} = \lim_{j\to\infty} \frac{(y - r_0)\cdot w^j}{|\vec{x} - \vec{r}|\cdot w^j} = \lim_{j\to\infty} \frac{y^{(j)} - r_0}{\left|\vec{x}^{(j)} - \vec{r}\right|} \in \left(\mathbf{d}f_{(r_0)}(\vec{r})\right)\left(\lim_{j\to\infty} \frac{\vec{x}^{(j)} - \vec{r}}{\left|\vec{x}^{(j)} - \vec{r}\right|}\right) =$$
$$\left(\mathbf{d}f_{(r_0)}(\vec{r})\right)\left(\lim_{j\to\infty} \frac{(\vec{x} - \vec{r})\cdot w^j}{|\vec{x} - \vec{r}|\cdot w^k}\right) = \left(\mathbf{d}f_{(r_0)}(\vec{r})\right)\left(\frac{\vec{x} - \vec{r}}{|\vec{x} - \vec{r}|}\right) = \frac{\left(\mathbf{d}f_{(r_0)}(\vec{r})\right)(\vec{x} - \vec{r})}{|\vec{x} - \vec{r}|}$$

out of which a formula for y can be extracted: $y \in \left(\mathbf{d}f_{(r_0)}(\vec{r})\right)(\vec{x} - \vec{r}) + r_0$. Since $(\vec{x}, y) \in \mathrm{Gr}(f)$ have been chosen arbitrarily, we can conclude that

$$f(\vec{x}) \subseteq \left(\mathbf{d}f_{(r_0)}(\vec{r})\right)(\vec{x} - \vec{r}) + r_0 \quad \text{for all } \vec{x} \in \textstyle\prod_{i\in P} I_i. \qquad \square$$

Now we will characterize the functions which have all of their unary instantiations affine. Although they themselves do not need to be affine, we will show that they are polynomials of a certain form. (This characterization is simple and well-known.)

Definition 4.1.18 (Multi-affine polynomials).
The n-ary polynomials over the variables $x_1, \dots, x_n$ of the form

$$\sum_{P\subseteq\{1,\dots,n\}} a_P \textstyle\prod_{i\in P} x_i$$

where all a_P are real numbers, are called *multi-affine*. Multi-affine polynomials of the form $a_1x_1 + \dots + a_nx_n + b$ are called *affine*.

The following classical property of multi-affine polynomials is the reason why they are important to us.

Lemma 4.1.19 (Detecting multi-affine polynomials).
Let $f\colon I_1 \times \dots \times I_n \to I_0$ *(*$n > 0$, $I_0, \dots, I_n \in \mathcal{I}$*) be a function such that for every* $j \in \{1, \dots, n\}$ *and every tuple* $X = (x_i)_{i\neq j} \in \prod_{i\neq j} I_i$*, the function* $f_{j,X}$ *is affine. Then* f *is a multi-affine polynomial.*

Proof. Let us proceed by induction on n. For $n = 1$ the claim is trivial. Suppose it holds for all $n < k$ (> 1) and prove it for $n = k$. For each $x \in I_n$ and $(x_1, \ldots, x_{n-1}) = X \in I_1 \times \cdots \times I_{n-1}$, define the functions $g_x \colon I_1 \times \cdots \times I_{n-1} \to I_0$ and $h_X \colon I_n \to I_0$ by $g_x(X) = h_X(x) = f(X, x)$. Notice that $h_X = f_{n,X}$. By the induction hypothesis and the assumption on $f_{n,X}$ we get:

$$g_x(X) = \sum_{P \subseteq \{1,\ldots,n-1\}} a_P(x) \prod_{i \in P} x_i$$

$$h_X(x) = a(X) \cdot x + b(X)$$

where all a_P are functions of type $I_n \to \mathbb{R}$ and a, b are functions of type $I_1 \times \cdots \times I_{n-1} \to \mathbb{R}$. Since the two expressions above are equal for every x, X, by taking two different fixed $t, t' \in I_n$, we get

$$a(X) \cdot (t - t') = h_X(t) - h_X(t') = g_t(X) - g_{t'}(X) = \sum_{P \subseteq \{1,\ldots,n-1\}} (a_P(t) - a_P(t')) \prod_{i \in P} x_i$$

for every X, which means that a is a multi-affine polynomial. Now it follows that

$$b(X) = -a(X)x + \sum_{P \subseteq \{1,\ldots,n-1\}} a_P(x) \prod_{i \in P} x_i$$

is also a multi-affine polynomial. Altogether, $f(x, X) = h_X(x) = a(X) \cdot x + b(X)$ is a multi-affine polynomial. ☐

We state a standard property of real polynomials which we will need in the proof to follow, i.e. that polynomials are locally determinable. This follows from the fact that a polynomial is either 0 or it has a finite set of roots.

Lemma 4.1.20 (Polynomials are locally determinable).
If two n-dimensional real polynomials agree on an open interval $J_1 \times \cdots \times J_n \subseteq \mathbb{R}^n$, then they are equal.

The main limitation theorem comes first in a version where the unary interval J from Proposition 4.1.13 is replaced by a multi-dimensional interval. Its corollary is a version in which this interval is replaced by an arbitrary open connected set.

Theorem 4.1.21. *Let $\mathcal{T}$ be a precise non-ignorant finite transducer of type $(\mathcal{S}_1 \times \cdots \times \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ with all representations $\mathcal{S}_i$ being affine. Any continuously differentiable thread $h \colon J \to I_0$ of $f_{\mathcal{T}}$ where $J = J_1 \times \cdots \times J_n \subset I_{\mathcal{S}_1} \times \cdots \times I_{\mathcal{S}_n}$ is an interval open in $\mathbb{R}^n$, is affine.*

Proof. For any $k \in \{1, \ldots, n\}$ and any set of values $x_i \in \mathrm{Fin}_{\mathcal{S}_i} \cap J_i$ for all $i \neq k$, the instantiation of $f_{\mathcal{T}}$ to the unary function $(f_{\mathcal{T}})_{k,(x_i)_{i \neq k}}$ is also finitely computable by Lemma 3.3.14 on page 45. Moreover, the analogous instantiation $h_{k,(x_i)_{i \neq k}} \colon J_i \to I_0$ of h is a thread of $(f_{\mathcal{T}})_{k,(x_i)_{i \neq k}}$. Therefore this unary thread is affine on J_k by Proposition 4.1.13. Since this holds for every $(x_i)_{i \neq k}$ in the dense set $\prod_{i \neq k} \mathrm{Fin}_{\mathcal{S}_i} \cap \prod_{i \neq k} J_i$, it holds for all values in $\prod_{i \neq k} J_i$. Therefore, by Lemma 4.1.19, h is a multi-linear polynomial on J.

Let us notice which of the arguments of h are not redundant and define a name for the set of their indices as follows:

$$P := \Big\{ k \in \{1, \dots, n\} \;\Big|\; \big(\exists \vec{x}, \vec{x}' \in J \text{ with } (i \neq k \Longrightarrow x_i = x_i')\big) \big(h(\vec{x}) \neq h(\vec{x}')\big) \Big\}.$$

Since the value of h within J does not depend on the value of an argument with an index $i \notin P$, the partial derivative g_i' for $i \notin P$ is constantly zero on the whole of J. However, for an argument index $i \in P$, g_i' can be zero only on a very limited part of J since it is a non-zero polynomial. Actually, the set $Z_i := \{\vec{x} \in J \,|\, g_i'(\vec{x}) = 0\}$ is a closed set in J with empty interior. (If it would have a non-empty interior, the polynomial g_i' would be constantly 0 on J by Lemma 4.1.20.) Therefore the finite union $\bigcup_{i \in P} Z_i$ is also closed and with empty interior. Its complement C is an open non-empty subset of J such that $f_i'(\vec{x})$ is not zero for every $i \in P$ and $\vec{x} \in C$.

Pick an open n-dimensional box $K = K_1 \times \dots \times K_n$ within the open set C. Take a point $\vec{x} \in K$ and some computation $\gamma \in \mathcal{H}_{\mathcal{T}}$ which inputs a representation of $\vec{x}$ (i.e. $[\![\Pi_{\mathrm{In}}(\gamma)]\!] = \vec{x}$) and outputs a representation of $h(\vec{x})$: $[\![\Pi_{\mathrm{Out}}(\gamma)]\!] = h(\vec{x})$.

Take a finite prefix $\gamma_1 \in \mathcal{H}_{\mathcal{T}}^{\mathrm{fin}}$ of γ such that $[\![\Pi_{\mathrm{In}}(\gamma_1)]\!] \subset K$ and $[\![\Pi_{\mathrm{In}}(\gamma_1)]\!] \times [\![\Pi_0(\gamma_1)]\!] \subset \mathrm{Zone}(h)$. This is possible because K and $\mathrm{Zone}(h)$ are open. Since $\mathcal{T}$ is a finite transducer, there must exist strings γ_2, γ_3, $\gamma_3 \neq \epsilon$, such that $\gamma_1\gamma_2, \gamma_1\gamma_2\gamma_3 \in \mathcal{H}_{\mathcal{T}}^{\mathrm{fin}}$ are prefixes of γ (i.e. $\gamma_1\gamma_2\gamma_3 \sqsubseteq \gamma$) which lead to the same state, i.e. $q_{\gamma_1\gamma_2} = q_{\gamma_1\gamma_2\gamma_3}$. Denote this state by q.

In this situation, we can apply the following consequence of Proposition 3.4.3

$$f_{\mathcal{T}}^{q} \subseteq \big(D_{\mathcal{S}_0}(\Pi_0(\gamma_1\gamma_2))\big)^{-1} \circ f_{\mathcal{T}} \circ \Big(\big(D_{\mathcal{S}_1}(\Pi_1(\gamma_1\gamma_2))\big), \dots, \big(D_{\mathcal{S}_n}(\Pi_n(\gamma_1\gamma_2))\big)\Big) \tag{4.8}$$

to conclude that $f_{\mathcal{T}}^{q}$ is single-valued because in (4.8) this function is dependent only on the thread h and not on the rest of $f_{\mathcal{T}}$. Another consequence of (4.8) is that the partial derivatives of $f_{\mathcal{T}}^{q}$ in coordinates in P are never zero because d-contractions have non-zero and non-infinite derivatives everywhere on their domain.

Now we can apply Lemma 4.1.17 on the following sub-self-similarity

$$f_{\mathcal{T}}^{q} \subseteq \big(D_{\mathcal{S}_0}(\Pi_0(\gamma_3))\big)^{-1} \circ f_{\mathcal{T}}^{q} \circ \big(\big(D_{\mathcal{S}_1}(\Pi_1(\gamma_3))\big), \dots, \big(D_{\mathcal{S}_n}(\Pi_n(\gamma_3))\big)\big)$$

with the set P to conclude that $f_{\mathcal{T}}^{q}$ is affine on all of its domain. This translates back to a part of the domain of h by formula (4.8). More precisely, h has been shown to be affine on $[\![\Pi_1(\gamma_1\gamma_2)]\!] \times \dots \times [\![\Pi_n(\gamma_1\gamma_2)]\!]$ which is within $K \subset J$. By Lemma 4.1.20 and the previously established fact that h is a multi-linear polynomial on J, it follows that h is affine on the whole of J. ☐

Corollary 4.1.22 (Main limitation theorem for affine transducers).
Let $\mathcal{T}$ be a precise non-ignorant finite transducer of type $(\mathcal{S}_1 \times \dots \times \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ with all representations $\mathcal{S}_i$ being affine. Any continuously differentiable thread $h\colon J \to I_0$ of $f_{\mathcal{T}}$ is affine.

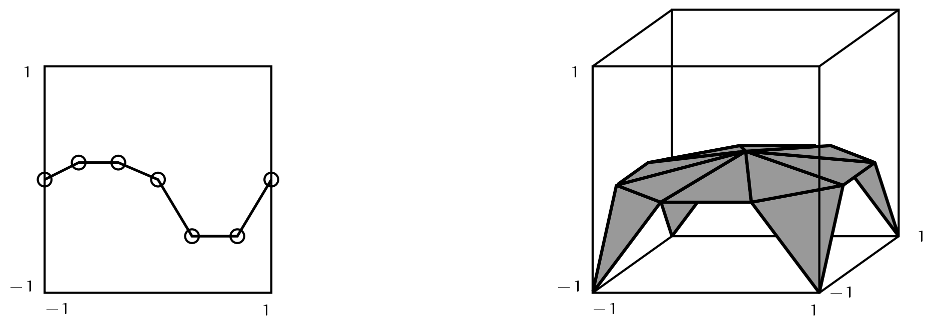

Figure 4.2: The graphs of two piecewise affine functions.

Proof. By Theorem 4.1.21, $h|_L$ is affine on each open n-dimensional open interval L within the open and connected set J. Such an interval surrounds every point in J since J is open. All we need to prove is that h coincides with the same affine function on every interval within J. Let $x, y \in J$ and $\xi\colon [0,1] \to J$ be a continuous injection with $\xi(0) = x$ and $\xi(1) = y$. For every $t \in [0,1]$, let L_t be an open interval within J that contains $\xi(t)$ and h coincides on it with an affine function $A_t\colon \mathbb{R}^n \to \mathbb{R}$. Put $\hat{t} = \sup\{t \in [0,1] \mid A_t = A_0\}$ and prove $\hat{t} = 1$ by contradiction. If $\hat{t} < 1$, then by continuity of ξ there is an $\varepsilon > 0$ such that $\xi([\hat{t}, \hat{t} + \varepsilon]) \subset L_{\hat{t}}$. It follows that $A_{\hat{t}+\varepsilon} = A_{\hat{t}} = A_0$, a contradiction to the choice of $\widehat{t}$.

We have shown that in the neighbourhood of any two points in J, the function h coincides with the same affine function. Therefore h is affine on J. □

4.2 Some positive results

In this section, we will show some of the real functions that are finitely computable using the signed binary representation. Among them are well-behaved functions (that have been proved locally affine) and functions that are not continuously differentiable anywhere. We will start by exploring some of the former ones.

4.2.1 Piecewise affine functions

These are continuous single-valued functions $f\colon V \to I_0$ (where $V = I_1 \times \cdots \times I_n$) whose graph is composed of n-dimensional polygons (see Figure 4.2). The aim of this section is to prove that whenever such a function f is of type $[-1,1]^n \to [-1,1]$ and the vertices of the polygons have rational coordinates (equivalently: the coefficients of the affine functions are rational), then there is a finite transducer computing f.

The opposite, i.e. that any piecewise affine function computable by finite transducer with signed binary representation has to have rational coefficients is also true. The unary case of this has been proved by Lisovik et al. in [1999]. The multi-dimensional version follows from the unary version by using Lemma 3.3.14. Thus, once we will have achieved the aim of this section, we will have obtained a complete characterization of piecewise affine functions computable by finite transducers with the signed binary representation.

Let us first define the notion of a piecewise affine function formally.

Definition 4.2.1 (Piecewise affine function).
A continuous function $f\colon V \to I_0$ is called *piecewise affine* if there exists a finite set $\mathcal{K}$ of n-dimensional polygons such that

$$(\forall C_1, C_2 \in \mathcal{K})(C_1^o \cap C_2^o = \emptyset) \quad \text{and} \quad \bigcup \mathcal{K} = V$$

and $f|_C$ is affine for each $C \in \mathcal{K}$.

Any piecewise affine function f *with rational coefficients* can be also denoted more specifically as follows:

$$f = \mathrm{Aff}\left(\mathcal{K}, \kappa, (a^{(j)})_{i\leqslant n}^{j\leqslant m}, (b^{(j)})^{j\leqslant m}, c\right)$$

where $m > 0$ is the number of different affine functions whose pieces may be used, $\kappa\colon \mathcal{K} \to \{1, \dots, m\}$ is an assignment of affine functions to the polygons, $a_i^{(j)}, b^{(j)} \in \mathbb{Z}$ for $1 \leqslant i \leqslant n$, $1 \leqslant j \leqslant m$, and $c \in \mathbb{N}$, $c > 0$, are coefficients such that for every $C \in \mathcal{K}$ it holds

$$f(\vec{x}) = \frac{a_1^{(\kappa(C))} x_1 + \dots + a_n^{(\kappa(C))} x_n + b^{(\kappa(C))}}{c} \tag{4.9}$$

for all $\vec{x} = (x_1, \dots, x_n) \in C$.

Sometimes by specifying all of the affine functions with which a piecewise-affine function f locally coincides, the function f is not uniquely determined. Nevertheless, there are only finitely many functions satisfying such a description by the following lemma.

Lemma 4.2.2 (Coefficients almost determine piecewise affine function).
Let $I_1, \dots, I_n \in \mathcal{I}$. For every collection of coefficients

- $m \in \mathbb{N}$, $m > 0$,
- $a_i^{(j)}, b^{(j)} \in \mathbb{Z}$ *for each* $1 \leqslant i \leqslant n$, $1 \leqslant j \leqslant m$ *and*
- $c \in \mathbb{N}$, $c > 0$

there are finitely many piecewise affine functions of the form $\mathrm{Aff}\left(\mathcal{K}, \kappa, (a_i^{(j)})_{i\leqslant n}^{j\leqslant m}, (b^{(j)})^{j\leqslant m}, c\right)$ *for some* $\mathcal{K}$ *and* κ.

Proof. There are altogether m possible affine functions that can be assigned to some polygons in $\mathcal{K}$. Let us denote them as follows for $j = 1, \dots, m$:

$$A_j(\vec{x}) = \frac{a_1^{(j)} x_1 + \cdots + a_n^{(j)} x_n + b^{(j)}}{c}.$$

Let us assume, without loss of generality, that these m functions are pairwise different. Let H_{j_1,j_2} be the set of solutions of the equation $A_{j_1} = A_{j_2}$. Each H_{j_1,j_2} is either a hyperplane in $\mathbb{R}^n$ or empty.

Any hyperplane H in $\mathbb{R}^n$ which contains the border of two adjacent polygons in $C_1, C_2 \in \mathcal{K}$ has the property that the affine functions $f|_{C_1}$ and $f|_{C_2}$ agree on H. Therefore H has to be identical to $H_{\kappa(C_1),\kappa(C_2)}$.

It follows that the boundary of each $C \in \mathcal{K}$ is formed solely by some of the hyperplanes among H_{j_1,j_2} and by the boundary hyperplanes of the n-dimensional interval V. Therefore C is a union of some of the sections of $\mathbb{R}^n$ cut out by all of the hyperplanes H_{j_1,j_2} within the interval V. Since there are not more than 2^m of these sections by Lemma 4.2.3 below and there are m affine functions to choose from in each section, there are not more than $m^{(2^m)}$ piecewise affine functions composed of pieces of the affine functions $A_1, \dots, A_m$. □

The following is a review of a basic combinatoric property of hyperplanes:

Lemma 4.2.3. *A set of m hyperplanes in $\mathbb{R}^n$ divide the whole space to not more than 2^m sections.*

Proof. Let us prove the claim by induction on m. With $m = 0$ there is exactly one section, i.e. the whole space $\mathbb{R}^n$. Thus the claim holds for $m = 0$. For $m > 0$, pick one of the hyperplanes H. By the induction hypothesis, the remaining $m - 1$ hyperplanes divide $\mathbb{R}^n$ to at most 2^{m-1} sections. The hyperplane H can divide each of these convex sections into at most two. Thus altogether the m hyperplanes divide $\mathbb{R}^n$ into at most $2 \cdot 2^{m-1} = 2^m$ sections. □

Another technical lemma establishes a bound on the coefficients b and on the total difference of the function, provided that there is a bound on the coefficients a.

Lemma 4.2.4. *For any piecewise affine function*

$$f = \mathrm{Aff}\left(\mathcal{K}, \kappa, (a_i^{(j)})_{i \leqslant n}^{j \leqslant m}, (b^{(j)})^{j \leqslant m}, c\right) : [-1, 1]^n \to [-1, 1]$$

with $\left|a_i^{(j)}\right| \leqslant c/4n$ *for all* i, j, *it holds*

1. *for every* $C \in \mathcal{K}$, $\quad \left|b^{(\kappa(C))}\right| \leqslant c \cdot \frac{5}{4}$
2. *for any* $\eth_1, \dots, \eth_n \in \{\bar{1}, 0, 1\}$ *and* $\vec{x}, \vec{y} \in (\llbracket \eth_1 \rrbracket, \dots, \llbracket \eth_n \rrbracket)$, $\quad |f(\vec{x}) - f(\vec{y})| \leqslant \frac{1}{2}$.

Proof. Take any point $\vec{x} \in C \in \mathcal{K}$ and put $j := \kappa(C)$. Since $f(\vec{x}) \in [-1, 1]$, it holds

$$\left|a_1^{(j)}x_1 + \cdots + a_n^{(j)}x_n + b^{(j)}\right| \leqslant c.$$

From this and $\left|a_i^{(j)}\right| \leqslant c/4n$, $|x_i| \leqslant 1$ and the triangular inequality, we get

$$c \ \geqslant \ \left|a_1^{(j)}x_1 \ + \ \cdots \ + \ a_n^{(j)}x_n \ + \ b^{(j)}\right| \ \geqslant \ \left|b^{(j)}\right| - \left|-a_1^{(j)}\right| - \cdots - \left|-a_n^{(j)}\right| \ \geqslant \ \left|b^{(j)}\right| - c/4$$

which yields $\left|b^{(j)}\right| \leqslant c + c/4 = c \cdot \frac{5}{4}$.

In order to prove the second statement, consider the points

$$\vec{z}^{(i)} = (x_1, \ldots, x_i, y_{i+1}, \ldots, y_n)$$

for all $i = 0, \ldots, n$. It holds $\vec{x} = \vec{z}^{(0)}$ and $\vec{y} = \vec{z}^{(n)}$. We will show that $\left|f(\vec{z}^{(i-1)}) - f(\vec{z}^{(i)})\right| \leqslant 1/2n$ for all $i = 1, \ldots, n$, from which the desired inequality will follow by the triangular inequality:

$$|f(\vec{x}) - f(\vec{y})| = \left|\sum_{1 \leqslant i \leqslant n} f(\vec{z}^{(i-1)}) - f(\vec{z}^{(i)})\right| \leqslant \sum_{1 \leqslant i \leqslant n} \left|f(\vec{z}^{(i-1)}) - f(\vec{z}^{(i)})\right| \leqslant \frac{1}{2}.$$

Consider the line connecting $\vec{z}^{(i-1)}$ and $\vec{z}^{(i)}$:

$$\left\{(x_1, \ldots, x_{i-1}, t, y_{i+1}, \ldots, y_n) \;\middle|\; x_i \leqslant t \leqslant y_i\right\}$$

where, without loss of generality, we assume $x_i \leqslant y_i$. Denote all of the points where this line leaves (for t going from x_i to y_i) some polygon in $\mathcal{K}$ by $\vec{p}_1, \ldots, \vec{p}_{s-1}$ and let $\vec{p}_0 := z^{(i-1)}$, $\vec{p}_s := z^{(i)}$. Also let $\beta\colon \{1, \ldots, s\} \to \{1, \ldots, m\}$ be the function assigning to each segment $\vec{p}_{\ell-1}, \vec{p}_\ell$ of this line an affine function with which f coincides in this segment. Now we can get what we need by the triangular inequality and $\left|\vec{z}^{(i-1)} - \vec{z}^{(i)}\right| \leqslant 2$:

$$\begin{aligned}\left|f(\vec{z}^{(i-1)}) - f(\vec{z}^{(i)})\right| &\leqslant \sum_{1 \leqslant \ell \leqslant s} \left|f(\vec{p}_{\ell-1}) - f(\vec{p}_\ell)\right| = \\ &\sum_{1 \leqslant \ell \leqslant s} \left(\frac{\left|a_i^{(\beta(\ell))}\right|}{c} \cdot \left|\vec{p}_{\ell-1} - \vec{p}_\ell\right|\right) \leqslant \frac{1}{4n} \cdot \sum_{1 \leqslant \ell \leqslant s} \left|\vec{p}_{\ell-1} - \vec{p}_\ell\right| = \frac{1}{4n} \cdot \left|\vec{z}^{(i-1)} - \vec{z}^{(i)}\right| \leqslant \frac{1}{2n}.\end{aligned}$$

□

Now we are ready to state and prove the promised theorem. We choose to prove it by a direct construction of the transducer. One could certainly make a good use of some closure properties of finitely computable functions and some characterization of piecewise affine functions like Lisovik et al. [1999] to get a clearer and slower-paced proof. Nevertheless, since studying such properties per se is out of the scope of this thesis, we choose the direct proof because it is shorter.

Theorem 4.2.5 (Signed binary computes piecewise affine functions).
For every piecewise affine function $\widehat{f}\colon [-1, 1]^n \to [-1, 1]$ *with rational coefficients, it holds* $\widehat{f} = f_{\mathcal{T}}$ *for a finite transducer* $\mathcal{T}\colon (\mathcal{S}_{\text{sigbin}}, \ldots, \mathcal{S}_{\text{sigbin}}) \to \mathcal{S}_{\text{sigbin}}$.

Proof. Let $\mathrm{Aff}\left(\widehat{\mathcal{K}}, \widehat{\kappa}, (\widehat{a}_i^{(j)})_{i\leqslant n}^{j\leqslant m}, (\widehat{b}^{(j)})^{j\leqslant m}, \widehat{c}\right)$ be a more specific name for $\widehat{f}$. Choose $k \in \mathbb{N}$, $k > 0$, such that $2^{-k} \cdot |\widehat{a}_i^{(j)}/\widehat{c}| < \frac{1}{4n}$ for every i, j.

Now we can define $\mathcal{T} = (Q, \zeta, q_I)$ as follows:

- $Q = Q_1 \cup Q_2$ where

$$Q_1 = \left\{ \widehat{f} \circ \left(D_{\text{sigbin}}(u_1), \dots, D_{\text{sigbin}}(u_n)\right) \;\middle|\; u_1, \dots, u_n \in \{\bar{1}, 0, 1\}^{k'},\ 0 \leqslant k' < k \right\}$$

$$Q_2 = \left\{ \mathrm{Aff}\left(\mathcal{K}, \kappa, (\widehat{a}_i^{(j)})_{i\leqslant n}^{j\leqslant m}, (b^{(j)})_{j\leqslant m}, 2^k \cdot \widehat{c}\right) \;\middle|\; \text{for all permissible } \mathcal{K}, \kappa, b^{(j)} \right\}$$

- $q_I = \widehat{f}$

- for $f \in Q_1 \setminus Q_2$,

$$\zeta(f) = \left\{ \left(f \circ \left(D_{\text{sigbin}}(\partial_1), \dots, D_{\text{sigbin}}(\partial_n)\right), \partial_1, \dots, \partial_n, \epsilon \right) \;\middle|\; \partial_1, \dots, \partial_n \in \{\bar{1}, 0, 1\} \right\}$$

- for $f \in Q_2$,

$$\zeta(f) = \Big\{ \left(D_{\text{sigbin}}(\partial_0)^{-1} \circ f \circ \left(D_{\text{sigbin}}(\partial_1), \dots, D_{\text{sigbin}}(\partial_n)\right), \partial_1, \dots, \partial_n, \partial_0 \right) \;\Big|\; \partial_0, \partial_1, \dots, \partial_n \in \{\bar{1}, 0, 1\},\ [\![\partial_0]\!] \supseteq f\left([\![\partial_1]\!], \dots, [\![\partial_n]\!]\right) \Big\}$$

We need to prove that the above tuple is a precise finite transducer and that it fulfils the premises of Proposition 3.4.4 on page 46 to conclude that it computes $\widehat{f}$. We will divide the proof to subgoals which we will state in italics.

1. Q *is finite.* The set Q_1 has $\leqslant 3^k$ elements. The set Q_2 is finite by Lemma 4.2.2 because there are only finitely many choices for $b^{(j)}$'s by Lemma 4.2.4.

2. The initial state is defined correctly because $f \in Q_1$ for $k' = 0$.

3. All $f \in Q$ *are of the form:*

$$\mathrm{Aff}\left(\mathcal{K}, \kappa, (\widehat{a}_i^{(j)})_{i\leqslant n}^{j\leqslant m}, b^{(j)}, 2^{\ell} \cdot \widehat{c}\right) \tag{4.10}$$

for some $\mathcal{K}$, κ, $b^{(j)}$ *and* $0 \leqslant \ell \leqslant k$. The functions in Q_2 are of this form by definition. The proof that also functions from $Q_1 \setminus Q_2$ belong to this category is by induction on k'. For $k' = 0$, $q_I = \widehat{f}$ is of the form (4.10) by definition. The induction step coincides with the proof of the next point because all states in Q_1 are accessible from q_I.

4. ζ *is defined correctly for* $f \in Q_1 \setminus Q_2$. Assume that f is of the form (4.10). Take any $\partial_1, \dots, \partial_n \in \{\bar{1}, 0, 1\}$ and show that[2] $f' := f \circ \left(D_{\text{sigbin}}(\partial_1), \dots, D_{\text{sigbin}}(\partial_n)\right)$ is equal to

$$\mathrm{Aff}\left(K', \kappa', (a_i^{(j)})_{i\leqslant n}^{j\leqslant m}, (b'^{(j)})^{j\leqslant m}, 2^{\ell'} \cdot c\right). \tag{4.11}$$

[2]The symbol f' does not refer to the derivative of f here.

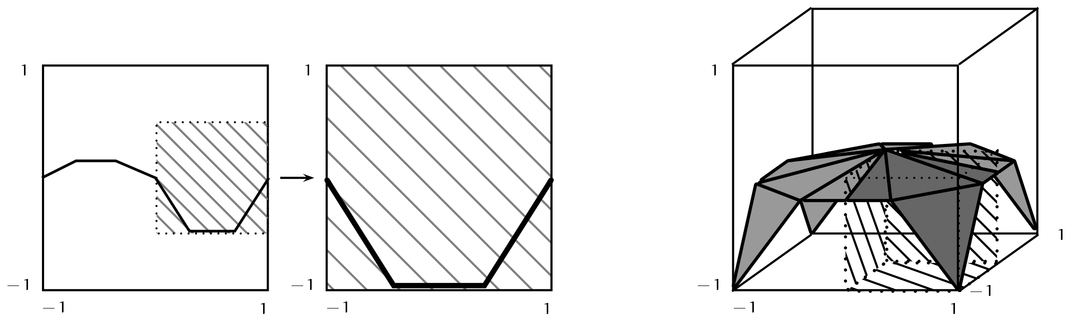

Figure 4.3: Clipping and enlarging piecewise affine functions during a transition.

for some $\mathcal{K}'$, κ', $b'^{(j)}$ and $0 \leqslant \ell' \leqslant k$.

To this end, we need to observe that an intersection of an n-dimensional interval (in this case $[\![\mathfrak{d}_1]\!] \times \cdots \times [\![\mathfrak{d}_n]\!]$) and a polygon is a (possibly empty) set of polygons. Therefore let $\mathcal{K}'$ be the version of $\mathcal{K}$ clipped by this interval and enlarged by the inverse of $(D_{\text{sigbin}}(\mathfrak{d}_1), \ldots, D_{\text{sigbin}}(\mathfrak{d}_n))$:

$$\mathcal{K}' := \left\{ \left(D_{\text{sigbin}}(\mathfrak{d}_1)^{-1}, \ldots, D_{\text{sigbin}}(\mathfrak{d}_n)^{-1} \right) (C \cap [\![\mathfrak{d}_1]\!] \times \cdots \times [\![\mathfrak{d}_n]\!]) \,\middle|\, C \in \mathcal{K} \right\} \setminus \{\emptyset\}.$$

On each polygon $C' \in K'$ which is an enlarged section of an old polygon $C \in \mathcal{K}$, we put $\kappa'(C') = \kappa(C)$ because

$$f'(\vec{x}) = \frac{\widehat{a}_1^{(\kappa(C))} \left(\frac{x_1 + \mathfrak{d}_1}{2}\right) + \cdots + \widehat{a}_n^{(\kappa(C))} \left(\frac{x_n + \mathfrak{d}_n}{2}\right) + b^{(\kappa(C))}}{\widehat{c}} \tag{4.12}$$

for $\vec{x} \in C'$, i.e. the coefficients of $x_1, \ldots, x_n$ are all halved. Therefore, by setting $\ell' := \ell + 1$ and $b'^{(j)}$ according to (4.12), we get a collection of coefficients that make (4.11) equal to f'.

5. ζ is defined correctly for $f \in Q_2$: This case is similar to the preceding one with the only difference that $\ell = k$ and that $f' := D_{\text{sigbin}}(\mathfrak{d}_0)^{-1} \circ f \circ \left(D_{\text{sigbin}}(\mathfrak{d}_1), \ldots, D_{\text{sigbin}}(\mathfrak{d}_n)\right)$ satisfies

$$f'(\vec{x}) = 2 \cdot \frac{\widehat{a}_1^{(\kappa(C))} \left(\frac{x_1 + \mathfrak{d}_1}{2}\right) + \cdots + \widehat{a}_n^{(\kappa(C))} \left(\frac{x_n + \mathfrak{d}_n}{2}\right) + b^{(\kappa(C))}}{\widehat{c}} - \mathfrak{d}_0. \tag{4.13}$$

Therefore f' is of the form (4.11) with $\ell' := \ell = k$.

The relationship of f' towards f can be visualized as follows: The graph of f is clipped by $[\![\mathfrak{d}_1]\!] \times \cdots \times [\![\mathfrak{d}_n]\!]$ and enlarged by the inverse of $(D_{\text{sigbin}}(\mathfrak{d}_1), \ldots, D_{\text{sigbin}}(\mathfrak{d}_n), D_{\text{sigbin}}(\mathfrak{d}_0))$ (see Figure 4.3).

6. $\mathcal{T}$ is total. This will follow from the fact that for every $f \in Q$ and for every $\mathfrak{d}_1, \ldots, \mathfrak{d}_n \in \{\bar{1}, 0, 1\}$ there are f' and s_0 such that

$$\left(f', \mathfrak{d}_1, \ldots, \mathfrak{d}_n, s_0\right) \in \zeta(f). \tag{4.14}$$

If $f \in Q_1 \setminus Q_2$, then this holds for $s_0 = \epsilon$ by the definition of ζ. If $f \in Q_2$, then we need to consider Lemma 4.2.4, point 2, to deduce that for any $\vec{x}, \vec{y} \in ([\![\mathfrak{d}_1]\!], \ldots, [\![\mathfrak{d}_n]\!])$ it holds $|f(\vec{x}) - f(\vec{y})| \leqslant$

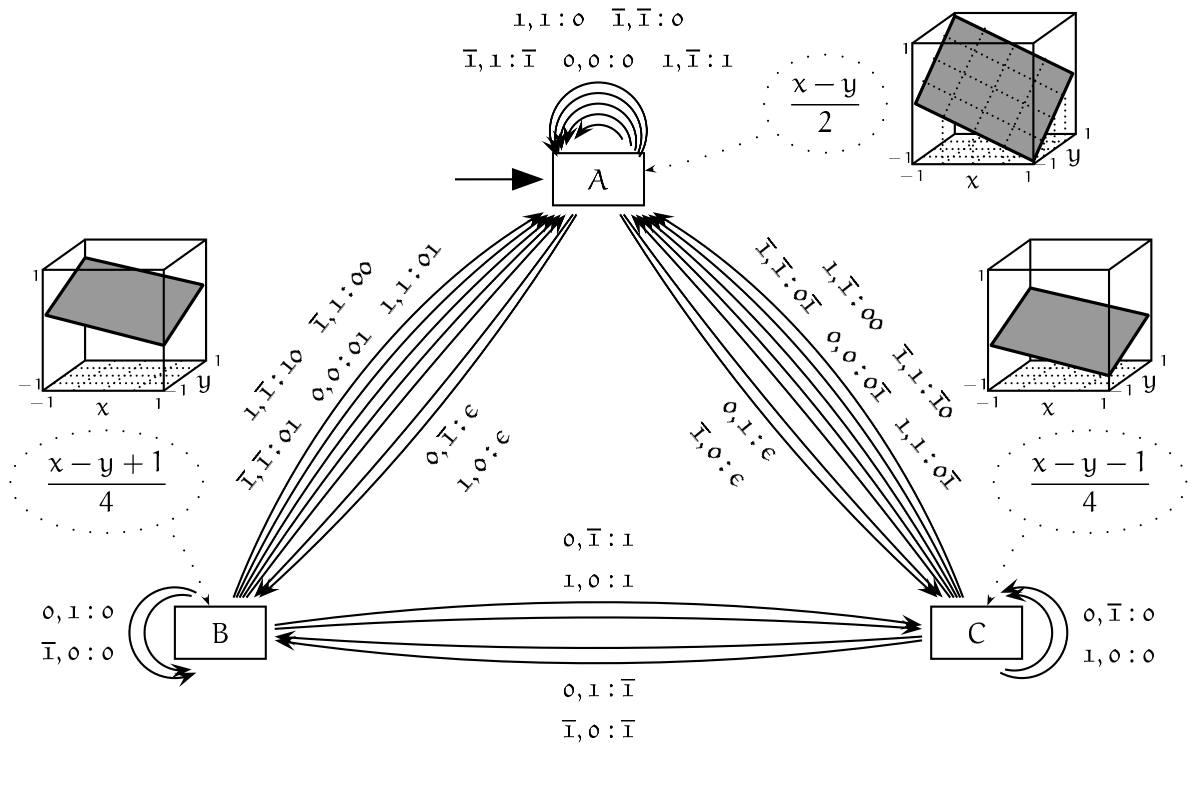

Figure 4.4: A finite automaton computing $(x - y)/2$.

$\frac{1}{2}$. From this we can deduce that $\left|f(\llbracket \mathfrak{d}_1 \rrbracket, \dots, \llbracket \mathfrak{d}_n \rrbracket)\right| \leqslant \frac{1}{2}$ and therefore there is a digit $\mathfrak{d}_0 \in \{\bar{1}, 0, 1\}$ which covers the whole range $f(\llbracket \mathfrak{d}_1 \rrbracket, \dots, \llbracket \mathfrak{d}_n \rrbracket)$. Thus for $s_0 := \mathfrak{d}_0$, there is f' such that (4.14) holds.

7. $\mathcal{T}$ is talkative. By Lemma 3.3.3 on page 38, it is sufficient to prove that there is no cycle without output in $\mathcal{T}$. The only transitions without output are those from states in $Q_1 \setminus Q_2$. There cannot be a cycle using only states from Q_1 because from a state $q \in Q_1$, independently from what the input is, the transducer $\mathcal{T}$ proceeds to a state in Q_2 within at most k steps by equation (4.12).

8. $\mathcal{T}^q$ computes q. We have finished the proof that $\mathcal{T}$ is a precise finite transducer. In order to be able to apply Proposition 3.4.4 with h assigning to each state itself as a function, we need to realize that formula (3.8) is trivially satisfied as an equation by $\mathcal{T}$ with such an assignment. ☐

As another example illustrating the process of creating a transducer which computed a given function, consider the finite automaton for the affine binary function $(x - y)/2$ shown in Figure 4.4 with the graphs of all its functions as states drawn next to them.

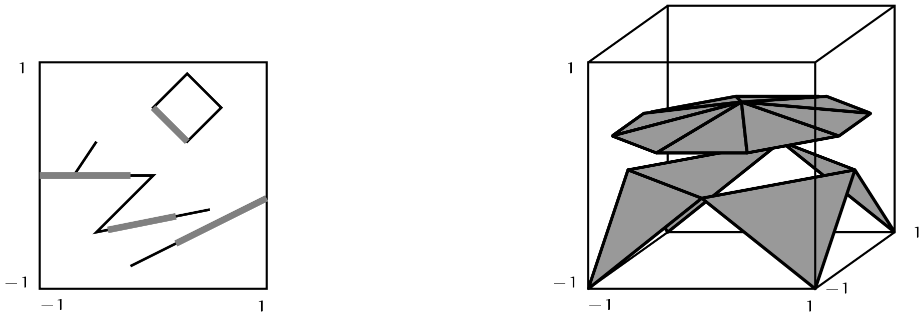

Figure 4.5: The graphs of two many-valued piecewise affine functions.

4.2.2 Many-valued piecewise affine functions

Many-valued piecewise affine functions are total functions whose graph is a union of the graphs of *partial* piecewise affine functions (see Figure 4.5 for an illustration). Among these we want to characterize those that are computable by a precise finite transducer. Let us first define the notion formally.

Definition 4.2.6 (Many-valued piecewise affine function).
A function $f\colon V \rightrightarrows I_0$ for which there is a finite covering $\mathcal{U} \subseteq \wp(V)$, a number $\varepsilon > 0$ and a piecewise affine function $f_U\colon V \to I_0$ for each $U \in \mathcal{U}$, such that

- for each $\vec{x} \in V$, there exists $U \in \mathcal{U}$ with $B_\varepsilon(\vec{x}) \cap V \subset U$ where $B_\varepsilon(\vec{x}) := [x_1 - \varepsilon, x_1 + \varepsilon] \times \cdots \times [x_n - \varepsilon, x_n + \varepsilon]$
- and $f = \bigcup_{U \in \mathcal{U}} (f_U|_U)$,

is called a *many-valued piecewise affine function*. The function f is said to have rational coefficients if all of the functions f_U have rational coefficients.

Notice that we do not demand anything from the system of sets $\mathcal{U}$ apart from overlapping by a uniform margin. It does not make a difference for indeterministic computation how far exactly every U stretches as long as there is another U' covering its edge so that the machine can safely output $f_{U'}(\vec{x})$ instead of $f_U(\vec{x})$ when the $\vec{x}$ is already known to be near the edge of U.

Theorem 4.2.7. *Every many-valued piecewise affine function* $f\colon [-1,1]^n \to [-1,1]$ *with rational coefficients is computed by a finite transducer using the signed binary representation.*

Proof. Let $f = \bigcup_{U \in \mathcal{U}} (f_U|_U)$ be a many-valued piecewise affine function and let ε be such a positive number that for every $\vec{x} \in V$ there is $U \in \mathcal{U}$ with $B_\varepsilon(\vec{x}) \cap V \subset U$.

There is a number $k \in \mathbb{N}$ such that $\left|[\![v]\!]\right| < 2\varepsilon$ for every $v \in \{\bar{1}, 0, 1\}^k$. Therefore, for each n-tuple $\vec{v} \in \left(\{\bar{1}, 0, 1\}^k\right)^n$, there is $U_{\vec{v}} \in \mathcal{U}$ with $[\![\vec{v}]\!] \subset U_{\vec{v}}$.

Since $f_{U_{\vec{v}}} \circ D_{\mathrm{sigbin}}(\vec{v})$ is a piecewise affine function for every $\vec{v} \in \left(\{\bar{1}, 0, 1\}^k\right)^n$, by Theorem 4.2.5, it is computed by a finite transducer which we will call $\mathcal{T}_{\vec{v}} = (Q_{\vec{v}}, \zeta_{\vec{v}}, (q_I)_{\vec{v}})$.

Now we can construct a finite transducer $\mathcal{T} = (Q, \zeta, q_I)$ computing f as follows:

- $Q := \{q_I\} \cup \bigcup_{\vec{v} \in \left(\{\bar{1},0,1\}^k\right)^n} Q_{\vec{v}}$,
- $\zeta(q_I) := \left\{ \left((q_I)_{\vec{v}}, \vec{v}, \epsilon\right) \,\middle|\, \vec{v} \in \left(\{\bar{1}, 0, 1\}^k\right)^n \right\}$
- $\zeta(q) := \zeta_{\vec{v}}(q)$ for every $q \in Q_{\vec{v}}$.

which means that we put all transducers $\mathcal{T}_{\vec{v}}$ next to each other, add a new initial state q_I and a new transition for each $\vec{v}$ leading from the initial state q_I to the initial state of $\mathcal{T}_{\vec{v}}$ with input $\vec{v}$ and empty output. □

It is also interesting to ask which many-valued piecewise affine functions are *completely* computed by a finite transducer using the signed binary representation. From the proof above it seems likely that these are those with every $U \in \mathcal{U}$ being a hyper-polygon whose sides are formed by the hyperplanes in V given by formulas of the form $x_i = d$ for some $i \in \{1, \ldots, n\}$ and a *dyadic* number d. For an illustration, the thick grey lines in the graph in Figure 4.5 on the left denote a function completely computed by some transducer that (incompletely) computes the whole function. For another transducer computing the function, the thick lines could be at different places.

To complete the characterization, consider that it follows from the single-valued case that for any many-valued piecewise affine function $f = \bigcup_{U \in \mathcal{U}} (f_U|_U)$ computed by a finite transducer using the signed binary representation, there is a subcovering $\mathcal{U}' \subseteq \mathcal{U}$ such that $f = \bigcup_{U \in \mathcal{U}'} (f_U|_U)$ is a many-valued piecewise affine function *with rational coefficients*.

4.2.3 A finite transducer computing a misbehaved function

As we mentioned in the introduction, there are some examples in the literature, of finitely computable functions whose derivative exhibits some fractal-like properties. For example, in [Lisovik 1990], there is a finite R_1^1-transducer computing a functions which is nowhere differentiable. Here, we show one computing a (single-valued) function which has derivative on a dense set of points but its values contain mutually interspersed 0's and $+\infty$'s.

How to construct a transducer which is extensional and computes an everywhere non-affine function? One possible way to do it is by using a weakened version of the construction based on Proposition 3.4.4 on page 46. That construction can be described as follows. We want to compute $\widehat{f}\colon I_1 \to I_0$ using representations $\mathcal{S}_1$, $\mathcal{S}_0$ of I_1 and I_0, respectively. We somehow construct a finite set of functions derived from $\widehat{f}$ by the formula

$$f \subseteq D_{\mathcal{S}_0}(u_0)^{-1} \circ \widehat{f} \circ \left(D_{\mathcal{S}_1}(u_1), \ldots, D_{\mathcal{S}_n}(u_n)\right).$$

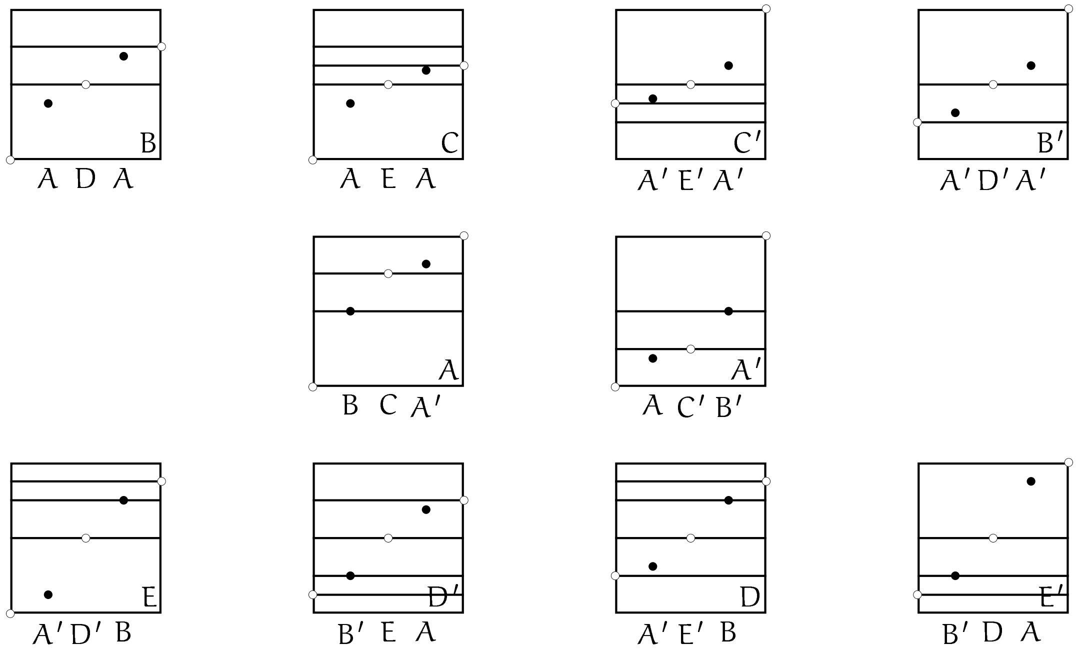

Figure 4.6: An intuitive idea of a derivation of the transducer below.

which is closed on enough transitions to form a total and talkative transducer.

In our case, we cannot start with any fully specified function $\widehat{f}$. Nevertheless, we can start with a partial specification of $\widehat{f}$, say, by specifying its values on a finite set of points. We do the same with the derived functions. To guarantee that the process will either finish or fail in finite time, we fix a finite set of arguments and require that a function in the generated set will be determined by its values for these arguments only.

To be more concrete, we can use $\mathcal{S}_0 = \mathcal{S}_1 = \mathcal{S}_{\mathrm{sigbin}}$ and take the set of arguments $\{-1, 0, 1\}$ as the determining set and start by stipulating that $\widehat{f}(-1) = -1$, $\widehat{f}(0) = 1/2$ and $\widehat{f}(1) = 1$. Thus if we succeed in the construction and moreover get a strongly connected extensional transducer, it should compute a function with the property 2 from Proposition 4.1.8 on page 53 and perhaps even more.

In this case, the construction can be successful and the sets of values for 10 derived functions in the three decisive points are illustrated in Figure 4.6. The letters underneath each graph denote the functions (using the notation from Figure 4.7) to which a transition can be made on inputting $\bar{1}$, 0 and 1, respectively. One can check that the resulting transducer will be extensional by computing another 2 values of each function in the arguments $-1/2$ and $1/2$ (these are indicated by black dots in the graphs in Figure 4.6). Each of these two can be derived in two different ways. Since both do agree for each function and each argument value, it is intuitively clear that the transitions will be extensional.

Example 4.2.8. Let $\mathcal{T}$ be the transducer in Figure 4.7. It is SC, extensional, of type $(\mathcal{S}_{\mathrm{sigbin}}) \to$

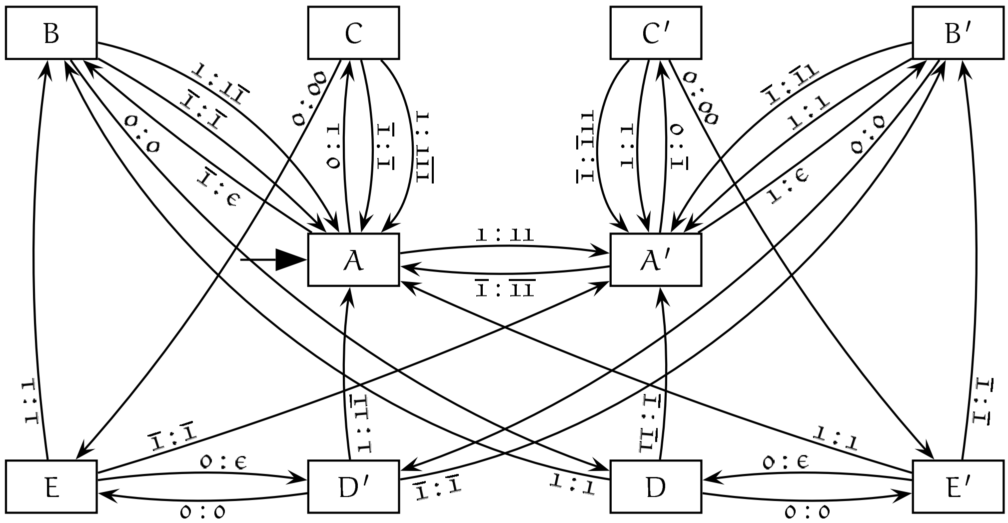

Figure 4.7: A finite transducer computing a function with a misbehaved derivative.

$\mathcal{S}_{\mathrm{sigbin}}$ and the single-valued function $f_{\mathcal{T}}$ is strictly increasing. Moreover, there are two dense sets $E_1, E_2 \subset [-1, 1]$ such that $f'_{\mathcal{T}}(x)$ exists for each $x \in E_1$ as well as for each $x \in E_2$ and it holds $x \in E_1 \implies f'_{\mathcal{T}}(x) = 0$ and $x \in E_2 \implies f'_{\mathcal{T}}(x) = +\infty$.

Proof outline. The transducer $\mathcal{T}$ is SC because there is a Hamiltonian cycle $A \to C \to E \to B \to D \to A' \to C' \to E' \to B' \to D' \to A$. It is total and deterministic because from each state there are exactly 3 transitions, with inputs $\bar{1}$, 0, 1, respectively. It is talkative because there are only four transitions with empty output (between $A \to B$, $A' \to B'$, $E \to D'$ and $E' \to D$), and these cannot form a cycle since no two of them even share a common state.

We will prove extensionality by observing that from each state, after inputting 01, the transducer will be in the same state and will have output an equivalent sequence like after inputting $1\bar{1}$, and the same for the dual pair $0\bar{1}$ versus $\bar{1}1$.[3] All the other equalities that $\mathcal{T}$ has to respect, i.e.

$$
\begin{array}{llll}
01^{k+1} = 10^k\bar{1} & 01^\infty = 10^\infty & 0\bar{1}^{k+1} = \bar{1}0^k1 & 0\bar{1}^\infty = \bar{1}0^\infty \\
\bar{1}1^{k+1} = 0^{k+1}\bar{1} & \bar{1}1^\infty = 0^\infty & 1\bar{1}^{k+1} = 0^{k+1}1 & 1\bar{1}^\infty = 0^\infty
\end{array}
$$

follow from $01 = 1\bar{1}$ and $0\bar{1} = \bar{1}1$. In the following table, we summarize to what state $\mathcal{T}$ will get and what output it will make when inputting either side of a particular equation from a particular state:

equation↓	A	B	C	D	E
$01 = 1\bar{1}$	$A/11\bar{1}\bar{1}$	$B/01 = 1\bar{1}$	$B/001 = 1\bar{1}\bar{1}$	$A/01 = 1\bar{1}$	$A/1\bar{1}$
$0\bar{1} = \bar{1}1$	$A/1\bar{1}$	$A'/0\bar{1}1 = \bar{1}11$	$A'/00\bar{1} = \bar{1}11$	$B'/0\bar{1} = \bar{1}1$	$B'/\bar{1}$

[3] Notice that this condition is sufficient but not necessary for a finite transducer of type $(\mathcal{S}_{\mathrm{sigbin}}) \rightrightarrows \mathcal{S}$ to be extensional.

Checking the equations for the states A'–E' is dual to the states A–E because all transitions from these 'dual' states have transitions leading to 'dual' states and with inputs and outputs negated when compared with their counterparts and also the two equations are duals of each other by negation.

Next we will prove that $f_{\mathcal{T}}$ is strictly increasing by the following technique: Each number within $[-1, 1]$ can be represented in the signed binary by a sequence of digits $\bar{1}$ and 1 only without the use of the digit 0. Moreover, the natural strict order of numbers is equivalent to the strict lexicographical ordering of such representations except for the fact that $[\![u\bar{1}1^\infty]\!] = [\![u1\bar{1}^\infty]\!]$ for every $u \in \{\bar{1}, 0, 1\}^*$. Thus it is sufficient to show the following for each state q:

1. After inputting 1, respective $\bar{1}$, the result will be in some set U^q_1, respective $U^q_{\bar{1}}$, with U^q_1 and $U^q_{\bar{1}}$ sharing only one point.
2. There is no way to obtain the output $11\cdots$ or $\bar{1}\bar{1}\cdots$ (which are the only representations of numbers 1 and -1) apart from the case when the very same stream is on input.

The following table shows the sets U^q_1 and $U^q_{\bar{1}}$ for all states modulo duality:

$\underrightarrow{q}$	A	B	C	D	E
$U^q_{\bar{1}}/U^q_1$	$[-1, \frac{1}{2}]/[\frac{1}{2}, 1]$	$[-1, 0]/[0, \frac{1}{2}]$	$[-1, 0]/[0, \frac{1}{4}]$	$[-1, 0]/[0, \frac{1}{2}]$	$[-1, 0]/[0, 1]$

To deduce that $f_{\mathcal{T}}$ has zero derivative on a dense set, it is crucial to observe that there are cycles with a higher speed of output than the speed of input, e.g. $A \to A' \to A$ with input $1\bar{1}$ and output $11\bar{1}\bar{1}$. It is also important that the fixpoint of the d-contraction corresponding to the total input on the cycle is not an endpoint of $[-1, 1]$. By Lemma 4.1.5, point 1b, this implies that the function computed in the states on such a cycle have a point with zero derivative. (If the cycle would represent an endpoint, it would only imply that one sided derivative is zero.) By Proposition 3.4.3, we can translate this property to any open subinterval of $[-1, 1]$ because $\mathcal{T}$ is SC. Therefore in any open subinterval, there is a point where $f_{\mathcal{T}}$ has derivative zero.

The analogous assertion about infinite derivative is dual because $f_{\mathcal{T}}$ is injective. It follows from Lemma 4.1.5, point 2b, since there is a cycle which has a faster input than the output, e.g. $A \to B \to D \to B \to A$ with input $\bar{1}01\bar{1}$ and output $01\bar{1}$. (Since $f_{\mathcal{T}}$ is increasing, it cannot have derivative equal to $-\infty$ anywhere.)

Chapter 5

Finite Automata and IFS-Representations

This chapter extends the limitation results from the previous chapter to a much wider class of IFS-representations. The method which we will use is based on a simple idea. As we already stated (but not proved) in Proposition 3.1.9 on page 29, any non-slow d-contraction can be viewed as an ordinary contraction using some metric. The proposition will be proved in this chapter. Moreover, it will follow that a non-slow d-contraction is even *affine* in a chosen metric. This is the crucial idea of our technique that any non-slow d-contraction can be viewed as an affine map.

There is a problem though that several different d-contractions might need a very different 'viewing angle' in order to be seen as affine maps. Therefore we cannot simply prove the generalizations of the results for affine IFS-representations using only the affine results themselves. We need to prove the generalizations all over again using generalized versions of the proofs from the previous chapter.

In case of many-valued functions, we cannot even reuse the proof technique of the affine version because it relies much on the properties of polynomials. Therefore we extend the proof technique from the unary affine case by adding some new, quite complex, features to handle multiple arguments.

5.1 Making d-contractions affine

Now we will formalize the idea of viewing a d-contraction as an affine contraction, i.e. 'stretching' a 'wavy' function to a straight one.

Definition 5.1.1. A d-contraction f on I is called *stretchable* if there exists an increasing continuously differentiable bijection $T\colon I \to I$ with $0 < T'(\mathrm{Fix}_f) < \infty$ such that $T \circ f \circ T^{-1}$ is an *affine* d-contraction on I. Every such a T is called a *stretching translator* of f. The affine function $T \circ f \circ T^{-1}$ is called a *stretched translation* of f.

In most situations, we will call a 'stretching translator' or a 'stretched translation' simply a 'translator' or a 'translation' because there are no other translators or translations that we could confuse them with, considered in this thesis.

Theorem 5.1.2. *Any d-contraction f on I that is not slow, is stretchable and has a unique stretching translator T. Moreover, this translator fulfils $0 < T'(x) < \infty$ for all $x \in I$.*

The theorem will be proved separately in Section 5.3. There, we will observe that the uniqueness relies on the conditions that T is increasing, continuously differentiable and its derivative in the fixpoint is not equal to 0 nor $+\infty$. This explains why we put these conditions to the definition of a translator. On the other hand, the proof of existence of such a smooth translator relies on the condition of double differentiability of a d-contraction.

Definition 5.1.3. Let us denote the unique stretching translator of a non-slow d-contraction f by $\mathbf{T_f}$. Put also $\mathbf{A_f} := T_f \circ f \circ T_f^{-1}$ to denote the unique stretched translation of f.

Obviously, the identity id_I is the translator of any affine d-contraction on I. For technical reasons, put also $\mathbf{T_{id_I}} := \mathrm{id}_I$ although id_I is not a d-contraction on I.

Next we will observe that stretching a d-contraction always preserves its strength. This explains why stretching does not work for slow d-contractions—there are no slow d-contractions among affine maps.

Lemma 5.1.4. *For any non-slow d-contraction f, it holds:* $\mathrm{Strg}_{A_f} = \mathrm{Strg}_f$.

Proof. Since f, T_f and T_f^{-1} are differentiable, we can use the chain rule as follows:

$$\begin{aligned}\mathrm{Strg}_{A_f} &= A_f'(\mathrm{Fix}_{A_f}) = (T_f \circ f \circ T_f^{-1})'(T_f(\mathrm{Fix}_f)) = \\ &T_f'(f(T_f^{-1}(T_f(\mathrm{Fix}_f)))) \cdot f'(T_f^{-1}(T_f(\mathrm{Fix}_f))) \cdot (T_f^{-1})'(T_f(\mathrm{Fix}_f)) = \\ &T_f'(\mathrm{Fix}_f) \cdot f'(\mathrm{Fix}_f) \cdot \frac{1}{T_f'(T_f^{-1}(T_f(\mathrm{Fix}_f)))} = T_f'(\mathrm{Fix}_f) \cdot \mathrm{Strg}_f \cdot \frac{1}{T_f'(\mathrm{Fix}_f)} = \mathrm{Strg}_f \qquad \square\end{aligned}$$

One of the corollaries of Theorem 5.1.2 is Proposition 3.1.9 about the relationship between d-contractions and contractions whose proof we still owe to the reader.

Proof of Proposition 3.1.9 on page 29. We have a non-slow d-contraction $f \in \mathrm{dContr}_I$ and we need to find a metric $\mu_f \colon I \times I \to [0, \infty)$ equivalent to $(x, y) \mapsto |x - y|$ such that f is a contraction with respect to it.

Put $\mu_f(x,y) := |T_f(x) - T_f(y)|$. Now it holds:

$$\mu_f\big(f(x), f(y)\big) = \big|T_f(f(x)) - T_f(f(y))\big| = \big|T_f\big(f\big(T_f^{-1}(T_f(x))\big)\big) - T_f\big(f\big(T_f^{-1}(T_f(y))\big)\big)\big| = \\ |A_f(T_f(x)) - A_f(T_f(y))| = \mathrm{Strg}_{A_f} \cdot |T_f(x) - T_f(y)| = \mathrm{Strg}_f \cdot \mu_f(x,y)$$

i.e. f is contracting relative to μ_f because $\mathrm{Strg}_f < 1$. This holds because f is non-slow. It remains to prove that μ_f is a metric on I, equivalent to the standard one. The following shows that μ_f is a metric:

1. $\mu_f(x,y) = 0 \Longleftrightarrow |T_f(x) - T_f(y)| = 0 \Longleftrightarrow T_f(x) = T_f(y) \Longleftrightarrow x = y$
2. $\mu_f(x,y) = |T_f(x) - T_f(y)| = |T_f(y) - T_f(x)| = \mu_f(y,x)$
3. $\mu_f(x,y) + \mu_f(y,z) = |T_f(x) - T_f(y)| + |T_f(y) - T_f(z)| \leqslant |T_f(x) - T_f(z)| = \mu_f(x,z)$

The translator T_f is continuously differentiable on a compact domain I. Therefore its derivative has a least and a largest value. Let us denote them as follows:

$$d := \min\left\{T'(x) \,\middle|\, x \in I\right\} \quad \text{and} \quad D := \max\left\{T'(x) \,\middle|\, x \in I\right\}.$$

By the mean value theorem, it holds $d \cdot |x-y| \leqslant |T(x) - T(y)| \leqslant D \cdot |x-y|$ for any $x, y \in I$. This implies that μ_f is equivalent to the Euclidean metric $(x,y) \mapsto |x-y|$. □

The existence of a translator for non-slow d-contractions allows us to prove the generalization of Lemma 3.1.6 on page 27 to arbitrary d-contractions. It turns out that iterated non-slow d-contractions converge exponentially like affine contractions while slow d-contractions converge sub-exponentially.

Proposition 5.1.5 (Strength is a measure of contractivity).
Let f *be a d-contraction on* I. *If* $\mathrm{Strg}_f < 1$ *then for every* $x, y \in I$, $x \neq y$, *there are real constants* $c, C > 0$ *such that*

$$c \cdot \mathrm{Strg}_f^k \leqslant \left|f^k(x) - f^k(y)\right| \leqslant C \cdot \mathrm{Strg}_f^k \quad \textit{for all } k \in \mathbb{N}.$$

If $\mathrm{Strg}_f = 1$, *then for every* $C > 0$ *and every* $0 < t < 1$, *there exists* $k \in \mathbb{N}$ *such that*

$$\left|f^k(x) - f^k(y)\right| > C \cdot t^k.$$

Proof. Assume first that $w := \mathrm{Strg}_f < 1$, i.e. that f is not slow. By Theorem 5.1.2, we can assume the existence of a unique stretching translator T_f.

Assuming w.l.o.g. that $x < y$, put $I_k := [f^k(x), f^k(y)]$ and $J_k := [T_f(f^k(x)), T_f(f^k(y))]$ for all $k \in \mathbb{N}$. By Lemma 3.1.6 and Lemma 5.1.4, it holds:

$$|J_k| = \left|T_f(f^k(x)) - T_f(f^k(y))\right| = \left|A_f^k(T_f(x)) - A_f^k(T_f(y))\right| = |T_f(x) - T_f(y)| \cdot \mathrm{Strg}_{A_f}^k = |J_0| \cdot w^k.$$

It will suffice to show that there are constants $r, R > 0$ and a neighbourhood Ω of Fix_f such that $r \cdot |T_f(U)| \leqslant |U| \leqslant R \cdot |T_f(U)|$ for all intervals $U \subseteq \Omega$ with $\mathrm{Fix}_f \in U$ because then

$$r \cdot |J_0| \cdot w^k = r \cdot |J_k| = r \cdot |T_f(I_k)| \leqslant |I_k| \leqslant R \cdot |T_f(I_k)| = R \cdot |J_k| = R \cdot |J_0| \cdot w^k.$$

for all sufficiently high k. Then one can put

$$c := \min\{r \cdot |J_0|, |I_0|, |I_1|/w, |I_2|/w^2, \ldots, |I_m|/w^m\}$$
$$C := \max\{R \cdot |J_0|, |I_0|, |I_1|/w, |I_2|/w^2, \ldots, |I_m|/w^m\}$$

where $k = 1, \ldots, m$ are all the indices for which I_k is not within Ω.

In order to prove the existence of such r, R, pick an $\varepsilon > 0$ and by the definition of continuous differentiability of f in Fix_f, get an open neighbourhood Ω such that $f'(x) \in [w - \varepsilon, w + \varepsilon]$ for all $x \in \Omega$. Put $\ell := w - \varepsilon$ and $L = w + \varepsilon$, assuming that ε had been chosen small enough to guarantee $\ell > 0$. For an arbitrary interval $U \subseteq \Omega$ it holds $\ell \cdot |U| \leqslant |f(U)| \leqslant L \cdot |U|$ because if there would be 2 points $x, y \in U$ with $|f(x) - f(y)| < \ell \cdot |x - y|$ or $|f(x) - f(y)| > L \cdot |x - y|$, then, by the mean value theorem, there would be $z \in U$ with $f'(z) < \ell$ or $f'(z) > L$ which contradicts the choice of Ω. Now we can put $r := 1/L$ and $R := 1/\ell$ to conclude the proof of the first half of the lemma.

Assume $\mathrm{Strg}_f = 1$. For *continuously* differentiable functions, it holds:

$$\lim_{k \to \infty} \frac{f^{k+1}(x) - f^{k+1}(y)}{f^k(x) - f^k(y)} = f'(\mathrm{Fix}_f) = 1$$

because both $f^k(x)$ and $f^k(y)$ converge to Fix_f for $k \to \infty$. Let $k_1 \in \mathbb{N}$ be such that

$$\frac{f^{k+1}(x) - f^{k+1}(y)}{f^k(x) - f^k(y)} \geqslant \frac{t+1}{2}$$

for all $k > k_1$. This yields

$$\left|f^{k_1+k_2}(x) - f^{k_1+k_2}(y)\right| \geqslant \left(\frac{t+1}{2}\right)^{k_2} \cdot \left|f^{k_1}(x) - f^{k_1}(y)\right|$$

for all $k_2 \in \mathbb{N}$. The right hand side is larger than $t^{k_1+k_2} \cdot C$ for a sufficiently large k_2 because it has a larger basis of exponentiation. Taking $k := k_1 + k_2$ with such a large k_2 proves the second part of the lemma. □

5.2 The limitation theorem

We will use a slightly different path to the theorem than we did in the affine case. First of all, the multi-dimensional version has to be proved much harder and without making use of the unary version. For this reason we will completely omit the unary version and aim straight for the multi-dimensional one. Like before, we will first prove the theorem for SC transducers, although with multiple input channels. Then we will use a method similar to the one in the unary affine finite case generalized to multiple input with non-slow d-contraction. It is mainly the generalization to multiple input which is to blame for the increase in complexity.

5.2.1 Consequences of non-slow sub-self-similarity

The following generalization of Lemma 4.1.5 on page 51 is formulated for non-slow d-contractions although the author conjectures that it also holds for arbitrary d-contractions. Although we will not need the lemma in any proofs in this thesis, it is nice and relevant enough to justify its place here.

Lemma 5.2.1 (Strength mismatch in sub-self-similarity).
In the situation of Definition 4.1.2 of the sub-self-similarity $f \subseteq g_0^{-1} \circ f \circ g_1$, assume also that $(r_1, r_0) \in \mathrm{Gr}\,(f)$ and that both of the contractions g_i are non-slow. Recall that we denote $r_i := \mathrm{Fix}_{g_i}$ and $w_i := \mathrm{Strg}_{g_i}$ for $i = 0, 1$.

1a. *If $w_1 > w_0$ (input is weaker than output), then the many-valued weak derivative $f^{\mathrm{wk}}_{(r_0)}(r_1)$ contains 0.*

1b. *If, moreover, f is single-valued, the ordinary derivative $f'(r_1)$ exists and is equal to 0.*

2a. *If $w_1 < w_0$ (input is stronger than output), then the many-valued weak derivative $f^{\mathrm{wk}}_{(r_0)}(r_1)$ contains 0 or ∞.*

2b. *If, moreover, f is single-valued and injective on some neighbourhood of r_1, the ordinary derivative $f'(r_1)$ exists and is equal to $+\infty$ or $-\infty$.*

Proof. In essence, we use the same technique in proving this lemma like for Lemma 4.1.5 but instead of using the exponential formula for the iterations of an affine d-contraction, we use the bounds for g_i with $i = 0, 1$, that follow from Proposition 5.1.5:

$$c_i \cdot w_i^k \leqslant [\![g_i^k]\!] \leqslant C_i \cdot w_i^k$$
$$m_i \cdot w_i^k \leqslant \min\left\{ |x - r_i| \,\middle|\, x \in [\![g_i^k]\!] \setminus [\![g_i^{k+1}]\!] \right\} \leqslant M_i \cdot w_i^k$$

which holds for some real constants $c_i, C_i, m_i, M_i > 0$ and for all $k \in \mathbb{N}$ and $i = 0, 1$. The former inequality follows from Proposition 5.1.5 with x, y equal to the endpoints of I_i and the latter one follows from double application of Proposition 5.1.5 with x, y equal to a_i, r_i and r_i, b_i, respectively, where $[a_i, b_i] := [\![g_i]\!]$. The values m_i and M_i are taken as the smaller one of the two lower bounds and the bigger one of the two upper bounds, respectively.

Choose any $x \in I_1 \setminus [\![g_1]\!]$ and $y \in f(x)$. Provided that $y \neq r_0$, let $k_1 \in \mathbb{N}$ be the only exponent such that $y \in [\![g_0^{k_1}]\!] \setminus [\![g_0^{k_1+1}]\!]$. Using this, we get the following estimate of a particular value of the weak derivative $f^{wk}_{(r_0)}(r_1)$:

$$\lim_{k\to\infty} \frac{c_0 \cdot w_0^k \cdot w_0^{k_1}}{M_1 \cdot w_1^k} \leqslant \lim_{k\to\infty} \frac{\min\left\{|x-r_0| \,\middle|\, x \in [\![g_0^{k+k_1}]\!] \setminus [\![g_0^{k+k_1+1}]\!]\right\}}{\left|[\![g_1^k]\!]\right|} \leqslant$$
$$\lim_{k\to\infty} \frac{\left|g_0^k(y)-r_0\right|}{\left|g_1^k(x)-r_1\right|} \leqslant \lim_{k\to\infty} \frac{\left|[\![g_0^k]\!]\right|}{\min\left\{|x-r_1| \,\middle|\, x \in [\![g_1^k]\!] \setminus [\![g_1^{k+1}]\!]\right\}} \leqslant \lim_{k\to\infty} \frac{M_0 \cdot w_0^k}{c_1 \cdot w_1^k}.$$

Both sides of this inequality are equal to 0 when $w_0 < w_1$, and to ∞ when $w_0 > w_1$. For $y = r_0$, the estimated limit trivially converges to 0. Thus the claims 1a and 2a have been proved.

The proofs of 1b and 2b in Lemma 4.1.5 can be used to prove 1b and 2b in this proposition with the only change that instead of the inequality (4.4), we need the following one:

$$\lim_{j\to\infty} \frac{|f(x_j)-r_0|}{|x_j-r_1|} \leqslant \lim_{j\to\infty} \frac{\left|[\![g_0^{k_j}]\!]\right|}{\min\{|x-r_1| \mid x \in [\![g_1^{k_j}]\!] \setminus [\![g_1^{k_j+1}]\!]\}} \leqslant \lim_{j\to\infty} \frac{C_0 \cdot w_0^{k_j}}{m_1 \cdot w_1^{k_j}} = 0$$

and instead of inequality (4.5) the following one:

$$\lim_{j\to\infty} \frac{|f(x_j)-r_0|}{|x_j-r_1|} \geqslant \lim_{j\to\infty} \frac{\min\{|x-r_0| \mid x \in [\![g_0^{k_j}]\!] \setminus [\![g_0^{k_j+1}]\!]\}}{\left|[\![g_1^{k_j}]\!]\right|} \geqslant \lim_{j\to\infty} \frac{m_0 \cdot w_0^{k_j}}{C_1 \cdot w_1^{k_j}} = \infty.$$

□

In the following, let $V = I_1 \times \cdots \times I_n$ for some $I_1, \ldots, I_n \in \mathcal{I}$, and let $I_0 \in \mathcal{I}$ as usual.

Lemma 5.2.2 (Multi-dimensional sub-self-similarity determines function).
In the situation of Definition 4.1.14 on page 58, if all of the contractions g_i are non-slow, the sub-self-similarity (4.7) implies:

If for some $P \subseteq \{1, \ldots, n\}$ the instantiation $\widehat{f} := f_{P,\vec{r}_{\overline{P}}}$ is differentiable in $\vec{r}_P$ and for every $k \in P$, the partial derivative $\left(f_{k,(r_i)_{i\neq k}}\right)'_{(r_0)}(r_k)$ does not include zero, then it holds:

$$\widehat{f}(\vec{x}) \subseteq T_{g_0}^{-1}\left(T'_{g_0}(r_0) \cdot \left(\mathbf{d}\widehat{f}_{(r_0)}(\vec{r}_P)\right)\left(\frac{T_{g_i}(x_i) - T_{g_i}(r_i)}{T'_{g_i}(r_i)}\right)_{i\in P} + T_{g_0}(r_0)\right)$$

for all $\vec{x} = (x_i)_{i\in P} \in \prod_{i\in P} I_i$.

In case when $\widehat{f}$ is single-valued, the formula above implies that $\widehat{f} = T_{g_0}^{-1} \circ A \circ T_{g_1}$ for some affine function $A\colon \prod_{i\in P} I_i \to I_0$.

Proof. In the same way like in the proof of Lemma 4.1.17 on page 59, we can assume without loss of generality that $P = \{1, \ldots, n\}$ and, consequently, $\widehat{f} = f$.

Since all g_i are not slow, each of them has a translator. Let us denote $T_i := T_{g_i}$, $A_i := A_{g_i}$ for $i = 0, \ldots, n$ and also $\vec{g} := (g_1, \ldots, g_n)$, $\vec{T} := (T_1, \ldots, T_n)$ and $\vec{A} := (A_1, \ldots, A_n)$.

The sub-self-similarity $f \subseteq g_0^{-1} \circ f \circ \vec{g}$ can be expressed as follows:

$$T_0 \circ f \circ \vec{T}^{-1} \subseteq T_0 \circ g_0^{-1} \circ T_0^{-1} \circ T_0 \circ f \circ \vec{T}^{-1} \circ \left(\vec{T} \circ \vec{g} \circ \vec{T}^{-1}\right) = A_0^{-1} \circ T_0 \circ f \circ \vec{T}^{-1} \circ \vec{A}$$

which is a self-similarity with affine d-contractions. Putting $h := T_0 \circ f \circ \vec{T}^{-1}$, we can apply Lemma 4.1.17 to get that

- either $\left(h_{k,(\mathrm{Fix}_{A_i})_{i\neq k}}\right)'_{(\mathrm{Fix}_{A_0})}(\mathrm{Fix}_{A_k})$ contains 0 for some $k \in \{1,\ldots,n\}$
- or $h(\vec{x}) \subseteq \left(\mathbf{d}h_{(\mathrm{Fix}_{A_0})}(\mathrm{Fix}_{\vec{A}})\right)(\vec{x} - \mathrm{Fix}_{\vec{A}}) + \mathrm{Fix}_{A_0}$ for all $\vec{x} \in V$.

The following equations show that the first of these two statements contradicts the conditions of this lemma and the second is equivalent to the inclusion which we are aiming at proving:

$$\mathrm{Fix}_{A_i} = T_i(r_i) \text{ for all } i = 0,\ldots,n \qquad \text{and}$$

$$\left(\mathbf{d}h_{(\mathrm{Fix}_{A_0})}(\mathrm{Fix}_{\vec{A}})\right)(\vec{t}) = \left(\mathbf{d}\left(T_0 \circ f \circ \vec{T}^{-1}\right)_{(T_0(r_0))}\left(\vec{T}(\vec{r})\right)\right)(\vec{t}) =$$

$$T_0'(r_0) \cdot \left(\mathbf{d}f_{(r_0)}\left(\vec{T}^{-1}(\vec{T}(\vec{r}))\right)\right)\left((T_1^{-1})'(T_1(r_1)) \cdot t_1, \ldots, (T_n^{-1})'(T_n(r_n)) \cdot t_n\right) =$$

$$T_0'(r_0) \cdot \left(\mathbf{d}f_{(r_0)}(\vec{r})\right)\left(\frac{t_1}{T_1'(r_1)}, \ldots, \frac{t_n}{T_n'(r_n)}\right)$$

where we applied the chain rule for many-valued functions from Lemma 2.6.7 and the normal rule for the derivative of the inverse of a function. □

5.2.2 Strongly connected transducers

Before we will be able to prove the limitation theorem for strongly connected transducers in a similar fashion like in the unary affine case, we will prepare a technique of dealing with the set P present in Lemma 5.2.2. Let us start by defining the maximal set P eligible for Lemma 5.2.2 as follows.

Definition 5.2.3 (Set of coordinates with non-zero derivative).
Let $f\colon V \to I_0$ be a continuously differentiable (single-valued) function. For any $\vec{x} \in V$, put

$$\mathbf{P}^f_{\vec{x}} := \left\{ i \in \{1,\ldots,n\} \,\middle|\, f_i'(\vec{x}) \neq 0 \right\}.$$

If this set is the same for all $\vec{x}$ in the domain of f, it is denoted simply by P^f and the function f is said to have a *constant sign of derivative*.

Lemma 5.2.2 does not tell us anything about the value of f for arguments in coordinates from $\overline{P}$ (denoting $P := P^f_{\vec{r}}$) different from $\vec{r}_{\overline{P}}$. But it fully determines the value of f for arguments $\vec{x}$ that coincide with $\vec{r}$ in coordinates from $\overline{P}$. It also holds that the partial derivatives of f in such an $\vec{x}$ are non-zero in all coordinates from P because the bijections T_{g_i} have non-zero and non-infinite derivative everywhere by Theorem 5.1.2 and A has the same differential everywhere because it is affine. Let us give a name to this property.

Definition 5.2.4 (Non-zero derivative spreading property).
A function $f\colon V \to I_0$ has the (non-zero derivative) *spreading property* in a point $\vec{x} \in V$ if for every $\vec{x}' \in V$ such that $(\vec{x})_{\overline{P}} = (\vec{x}')_{\overline{P}}$ where $P := P^f_{\vec{x}}$, it holds $P^f_{\vec{x}'} \supseteq P^f_{\vec{x}}$. In this case, the elements of the set $P^f_{\vec{x}}$ are called the (non-zero derivative) *spreading coordinates* of f in $\vec{x}$.

Now we can state formally what we have discussed before the definition as follows:

Corollary 5.2.5 (of Lemma 5.2.2).
In the situation of Definition 4.1.14, if all of the contractions g_i are non-slow, the sub-self-similarity (4.7) implies that f has the spreading property in $\vec{r}$.

In an SC transducer, we can expect to be able to find a set of sub-self-similarities whose centres are dense in V. We will show that for a continuously differentiable function, Lemma 5.2.2 can be applied to all of the sub-self-similarities with the *same* maximal P. The fact that there will be a dense set on which the partial derivatives in the coordinates from $\overline{P}$ are zero means that the continuously differentiable f will have partial derivatives in every $i \in \overline{P}$ equal to zero everywhere and thus be completely independent on the arguments in the coordinates from $\overline{P}$. The function f will then be completely determined by Lemma 5.2.2 for any of the sub-self-similarities mentioned above. The following lemma embodies the key part of this idea.

Lemma 5.2.6 (Global set of spreading coordinates).
Let $f\colon V \to I_0$ be a continuously differentiable *function. If there exists a dense set $E \subseteq V$ of points in which f has the spreading property, then f has a constant sign of derivative.*

Proof. Let $\vec{e} \in E$ be some point with maximal set of spreading coordinates, i.e. one for which there is no $\vec{x} \in E$ such that $P^f_{\vec{x}} \supsetneqq P^f_{\vec{e}}$. Denote $P := P^f_{\vec{e}}$.

First we will prove that P is maximal among $P^f_{\vec{x}}$ for all $\vec{x} \in V$, i.e. not only on E. Assume, on the contrary, that $P^f_{\vec{x}} \supsetneqq P$. By the continuity of the derivative, there has to be a neighbourhood U of $\vec{x}$, in which the partial derivatives f'_i for $i \in P^f_{\vec{x}}$ are still non-zero. Because E is dense, there is $\vec{e}' \in E \cap U$ for which it holds $P^f_{\vec{e}'} \supseteq P^f_{\vec{x}} \supsetneqq P$, a contradiction to the maximality of $P^f_{\vec{e}}$ within E.

Let us notice that $P^f_{\vec{x}} = P$ for all $\vec{x}$ with $(\vec{x})_{\overline{P}} = (\vec{e})_{\overline{P}}$. By definition of the spreading property for $\vec{e}$, it holds $P^f_{\vec{x}} \supseteq P$. The equality now follows from the maximality of P.

Next we will prove that $P^f_{\vec{x}} \subseteq P$ for any $\vec{x} \in V$. Close to $\vec{x}$, there is an $\vec{e}' \in E$ with $P^f_{\vec{x}} \subseteq P^f_{\vec{e}'}$. By the spreading property of $\vec{e}'$, it holds $P^f_{\vec{e}'} \subseteq P^f_{\vec{e}''}$ where $\vec{e}''$ is the point which agrees with $\vec{e}'$ on the coordinates from P and with $\vec{e}$ on the coordinates from $\overline{P}$. Since we have already noticed that $P^f_{\vec{e}''} = P$, we get $P^f_{\vec{x}} \subseteq P^f_{\vec{e}'} \subseteq P^f_{\vec{e}''} = P$.

The formula $P^f_{\vec{x}} \subseteq P$ means that $f'_i(\vec{x}) = 0$ for all $i \in \overline{P}$. Thus for every $(x_i)_{i \in P} \in \prod_{i \in P} I_i$, the function $f_{\overline{P},(x_i)_{i\in P}}$ is constant. This means that $f(\vec{x}) = f(\vec{x}')$ for any $\vec{x}, \vec{x}' \in V$ with $(\vec{x})_P = (\vec{x}')_P$, and subsequently $f'_i(\vec{x}) = f'_i(\vec{x}')$ for all $i \in P$. Since for every $\vec{x} \in V$ there is a $\vec{x}'$ with $(\vec{x})_P = (\vec{x}')_P$ and $(\vec{x}')_{\overline{P}} = (\vec{e})_{\overline{P}}$, it holds $f'_i(\vec{x}) \neq 0$ for every $i \in P$. Thus $P^f_{\vec{x}} = P$ for all $\vec{x} \in V$. ☐

Now we are ready to prove the limitation theorem for SC transducers with non-slow representations. We can strengthen the theorem a little bit by not demanding the representations to be non-slow when representing an endpoint of the base interval. This will turn out to be useful when considering LFT-representations later in Chapter 6.

Definition 5.2.7. A d-contraction $g\colon I \to I$ is *non-slow on the interior* if it is either non-slow or its fixpoint is an endpoint of I.

An IFS-representation (I, Σ, D) is *non-slow on the interior* if all d-contractions $D(u)$ for $u \in \Sigma^*$, are non-slow on the interior.

Proposition 5.2.8 (The main limitation theorem for SC transducers).
Let $\mathcal{T} = (Q, \zeta, q_I)$ be a precise non-ignorant strongly connected IFS-transducer of type $(\mathcal{S}_1, \dots, \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ with all representations $\mathcal{S}_0, \dots, \mathcal{S}_n$ non-slow on the interior. Then one of the following holds:

1. *$f_{\mathcal{T}}$ does not have any threads.*
2. *$f_{\mathcal{T}}$ is single-valued but not continuously differentiable on any open $U \subseteq I_{\mathcal{S}_1} \times \cdots \times I_{\mathcal{S}_n}$.*
3. *$f_{\mathcal{T}}$ is single-valued and equal to $T^{-1}_{D(\nu_0)} \circ A \circ (T_{D(\nu_1)}, \dots, T_{D(\nu_n)})$ for some $\nu_i \in \Sigma^*_{\mathcal{S}_i}$, for $i = 0, \dots, n$, and for some affine $A\colon I_{\mathcal{S}_1} \times \cdots \times I_{\mathcal{S}_n} \to I_{\mathcal{S}_0}$.*

Proof. Denote $V := I_{\mathcal{S}_1} \times \cdots \times I_{\mathcal{S}_n}$.

Assume that 1 and 2 are not true and aim at proving 3. Take a thread $h\colon J \to I_{\mathcal{S}_0}$ of $f_{\mathcal{T}}$ and assume w.l.o.g. that $J \subseteq V^o$, i.e. the open set J in V does not contain any points of the boundary of V. Like in the proof of Proposition 4.1.8 on page 53, we define

$$C_h := \left\{ \gamma \in \mathcal{H}^{\text{fin}}_{\mathcal{T}} \,\middle|\, q_\gamma = q_I,\ [\![\Pi_{\text{In}}(\gamma)]\!] \times [\![\Pi_0(\gamma)]\!] \subseteq \text{Zone}(h) \right\} \quad \text{and} \quad E_h := \{\vec{r}_\gamma \,|\, \gamma \in C_h\}$$

where $\vec{r}_\gamma := \left(r^{(1)}_\gamma, \dots, r^{(n)}_\gamma\right)$ for each $\gamma \in C_h$, and $r^{(i)}_\gamma := \text{Fix}_{D_{\mathcal{S}_i}(\Pi_i(\gamma))}$ for $i = 1, \dots, n$.

This means that C_h is a set of the computations that will lead the transducer $\mathcal{T}$ back to the initial state and at the same time, if iterated forever, would compute one of the values of the thread h. Notice that none of the intervals $[\![\Pi_i(\gamma)]\!]$ can contain an endpoint of $I_{\mathcal{S}_i}$ which means that each $D_{\mathcal{S}_i}(\Pi_i(\gamma))$ is non-slow.

Analogously like in the proof of Proposition 4.1.8, we can prove that E_h is dense in J and derive the following sub-self-similarity for each $\gamma \in C_h$:

$$f_{\mathcal{J}} \subseteq D_{\mathcal{S}_0}(\Pi_0(\gamma))^{-1} \circ f_{\mathcal{J}} \circ \left(D_{\mathcal{S}_1}(\Pi_1(\gamma)), \ldots, D_{\mathcal{S}_n}(\Pi_n(\gamma))\right) \tag{5.1}$$

This equation allows us to use Lemma 4.1.16 on page 59 to deduce that $f_{\mathcal{J}}$ is single-valued and assume $J = V^o$ and $h = (f_{\mathcal{J}})|_J$.

The negation of 2 implies that $f_{\mathcal{J}}$ is continuously differentiable on some open set U. We will prove that in this case, $f_{\mathcal{J}}$ is continuously differentiable everywhere. By Lemma 3.2.8 on page 32 and the proof of density of E_h, we can prove that there is $\gamma_U \in C_h$ such that $[\![\Pi_{In}(\gamma_U)]\!] \subseteq U$. In the formula (5.1) for γ_U in place of γ, which holds as equation, the right hand side is a continuously differentiable function. The left hand side, i.e. $f_{\mathcal{J}}$, is therefore also continuously differentiable.

By Corollary 5.2.5, $h = f_{\mathcal{J}}|_{V^o}$ has the spreading property in each point $\vec{r}_\gamma \in E_h$. Since $E_h \subseteq V^o$ is dense in V, by Lemma 5.2.6, f has constant sign of derivative. This means that for $P := P^f$, the instantiation $(f_{\mathcal{J}})_{\overline{P},(x_i)_{i\in P}}$ is constant for each $(x_i)_{i\in P} \in \prod_{i\in P} I_{\mathcal{S}_i}$. This statement together with Lemma 5.2.2 applied on equation (5.1) with any $\gamma \in C_h$ imply

$$f_{\mathcal{J}}(\vec{x}) = T_0^{-1}\left(T_0'(r_0) \cdot (\,\mathbf{d}f_{\mathcal{J}}(\vec{r}_\gamma))\left(\frac{T_1(x_1) - T_1(r_1)}{T_1'(r_1)}, \ldots, \frac{T_n(x_n) - T_n(r_n)}{T_n'(r_n)}\right) + T_0\left(f_{\mathcal{J}}(\vec{r}_\gamma)\right)\right)$$

where $r_i = r_\gamma^{(i)}$, $T_i = T_{D(v_i)}$ and $v_i = \Pi_i(\gamma)$. (Notice that for any $i \in \overline{P}$ the formula above does not depend on x_i which nevertheless appears in the formula.) Therefore by assigning to A the following affine function:

$$A(x_1, \ldots, x_n) = T_0'(r_0) \cdot (\,\mathbf{d}f_{\mathcal{J}}(\vec{r}_\gamma))\left(\frac{x_1 - T_1(r_1)}{T_1'(r_1)}, \ldots, \frac{x_n - T_n(r_n)}{T_n'(r_n)}\right) + T_0\left(f_{\mathcal{J}}(\vec{r}_\gamma)\right)$$

we have proved statement 3. □

5.2.3 Multiple input finite transducers

We will be proving the limitation theorem by induction on the depth of a vertex in the underlying graph like in the affine unary case in Subsection 4.1.2 on page 55. In the affine case, we relied on the fact that affine functions are locally determinable, i.e. within the class of all affine functions, a function can be identified solely by its values on an arbitrary open set. In other words, if two affine functions coincide on some open set, they are equal everywhere. This property cannot be expected from a rather arbitrary set of d-contractions, like the one which is generated by an arbitrary IFS-representation. Therefore we will need to formulate the theorem using one of the following closures.

Definition 5.2.9 (Gluing closure of function set).
Let $\mathcal{F}$ be a set of functions of type $J \to I_0$ where $J \subseteq V$.

A function $f\colon J \to I_0$ is called a *glued function* over $\mathcal{F}$ if there exists an *open* covering $\mathcal{U} \subseteq \wp(J)$ of J such that $f|_U \in \mathcal{F}|_U$ for every $U \in \mathcal{U}$.

The *gluing closure* of $\mathcal{F}$, denoted by *Glue*$(\mathcal{F})$ is the set of all glued functions over $\mathcal{F}$.

The graph of a function in the gluing closure of $\mathcal{F}$ can be imagined as being composed of many pieces of graphs of functions from $\mathcal{F}$ that are glued together by overlapping surfaces (more precisely: overlapping n-dimensional surfaces).

Remark 5.2.10. In the context of the above definition, it holds $\mathcal{F} \subseteq \mathit{Glue}(\mathcal{F})$. If $\mathcal{F}$ is locally determinable, then $\mathcal{F} = \mathit{Glue}(\mathcal{F})$.

In particular, the set of all affine functions is equal to its gluing closure.

Again, the main limitation theorem comes first in a version which applies to threads on a multidimensional interval. Its corollary is a version in which this interval is replaced by an arbitrary open connected set.

Theorem 5.2.11 (Main limitation theorem for threads on a rectangle).
Let $\mathcal{T}$ be a precise non-ignorant finite transducer of type $(\mathcal{S}_1 \times \cdots \times \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ with all representations $\mathcal{S}_i$ being open *and non-slow on the interior. Define the function set*

$$\mathcal{A} := \Big\{ D_{\mathcal{S}_0}(v_0)^{-1} \circ T_{D_{\mathcal{S}_0}(u_0)} \circ A \circ \Big(T_{D_{\mathcal{S}_1}(u_1)} \circ D_{\mathcal{S}_1}(v_1), \ldots, T_{D_{\mathcal{S}_n}(u_n)} \circ D_{\mathcal{S}_n}(v_n) \Big) \;\Big|\; A\colon V \to I_{\mathcal{S}_0} \text{ is affine},\ u_i, v_i \in \Sigma^*_{\mathcal{S}_i},\ \mathrm{Fix}_{u_i} \in I^o_{\mathcal{S}_i} \text{ for } i = 0, \ldots, n \Big\}.$$

Any continuously differentiable thread $h\colon J \to I_0$ of $f_{\mathcal{T}}$ where $J = J_1 \times \cdots \times J_n \subset V$ is an open interval, is in Glue$(\mathcal{A}|_J)$.

Before proving the theorem, we need to single out several simple properties of the set $\mathcal{A}$ and its closures and restrictions in the following lemmas.

Lemma 5.2.12. *Any function*

$$f = D_{\mathcal{S}_0}(v_0)^{-1} \circ T_{D_{\mathcal{S}_0}(u_0)} \circ A \circ \Big(T_{D_{\mathcal{S}_1}(u_1)} \circ D_{\mathcal{S}_1}(v_1), \ldots, T_{D_{\mathcal{S}_n}(u_n)} \circ D_{\mathcal{S}_n}(v_n) \Big)$$

from the set $\mathcal{A}$ from previous theorem has a constant sign of derivative and it holds $P^f = P^A$. If J is connected, the same holds for any $f \in$ Glue$(\mathcal{A}|_J)$.

Proof. By Theorem 5.1.2, all the translators $T_{D_{\mathcal{S}_i}(u_i)}$ are continuously differentiable with positive, finite derivative on the whole of $I_{\mathcal{S}_i}$. The chain rule and the rule for the derivative of an inverse can be applied to deduce that f is continuously differentiable. The sign of the partial derivative $f'_i(\vec{x})$ in an arbitrary $\vec{x} \in V$ coincides with the sign of the partial derivative A'_i in a particular point in V. Therefore the sign of $f'_i(\vec{x})$ does not depend on the choice of $\vec{x}$.

Let $f \in Glue(\mathcal{A}|_J)$ which means that there is an open covering $\mathcal{U}$ of V and an assignment $\kappa\colon \mathcal{U} \to \mathcal{A}|_J$ with $f|_U = \kappa(U)|_U$ for all $U \in \mathcal{U}$. We already know that each $f|_U$ has a constant sign of derivative.

Take any two points $\vec{x}, \vec{y} \in J$. By the connectivity of J, there is a continuous injective map $\xi\colon [0,1] \to J$ connecting x with y: $\xi(0) = \vec{x}$, $\xi(1) = \vec{y}$. Assume that for some $i \in \{1, \dots, n\}$, it holds $f_i'(x) = 0$. Define $t_M := \sup\{t \in [0,1] \mid f_i'(\xi(t)) = 0\}$ and $z := \xi(t_M)$. Let $U_z \in \mathcal{U}$ be an open set containing z. The function f has constant sign of derivative on U_z and therefore also on $L_z := U_z \cap \{\xi(t) \mid t \in [0,1]\}$. Since $z \in L_z$ and U_z is open, there is an $\varepsilon > 0$ such that $f_i' \circ \xi$ has constant sign on the interval $[t_M - \varepsilon, t_M + \varepsilon] \cap [0,1]$. By the definition of t_M, there is some $t \in [t_M - \varepsilon, t_M]$ with $f_i'(\xi(t)) = 0$. Thus $f_i' \circ \xi$ is 0 on the whole of $[t_M - \varepsilon, t_M + \varepsilon] \cap [0,1]$ which has to be equal to $[t_M - \varepsilon, t_M]$ because of the definition of t_M as a supremum. This is possible only when $t_M = 1$. Therefore also $f_i'(y) = f_i'(\xi(1)) = f_i'(\xi(t_M)) = 0$.

We showed that if the value of a partial derivative is equal to zero at any point of J, it implies that this partial derivative is zero in all points of J. This is a reformulation of f having constant sign of derivative on J. ☐

Lemma 5.2.13. *In the situation of Theorem 5.2.11, let $J \subseteq V$ and $\mathcal{K} \subseteq \wp(J)$. If every $K \in \mathcal{K}$ is open and the continuously differentiable function $h\colon J \to I_0$ fulfils $h|_K \in Glue(\mathcal{A}|_K)$ for every $K \in \mathcal{K}$, then also $h|_{\bigcup\mathcal{K}} \in Glue(\mathcal{A}|_{\bigcup\mathcal{K}})$.*

Proof. Assume that $h|_K \in Glue(\mathcal{A}|_K)$ for all $K \in \mathcal{K}$. Denote by $\mathcal{U}_K \subseteq \wp(K)$ an open covering of K for which it holds $h|_U \in (\mathcal{A}|_K)|_U$ for each $U \in \mathcal{U}_K$. Since $(\mathcal{A}|_K)|_U = \mathcal{A}|_U$, it holds $h|_U \in \mathcal{A}|_U$ for each $U \in \bigcup_{K\in\mathcal{K}} \mathcal{U}_K =: \mathcal{U}$. Since this $\mathcal{U}$ covers $\bigcup\mathcal{K}$, we can already conclude that $h|_{\bigcup\mathcal{K}} \in Glue(\mathcal{A}|_{\bigcup\mathcal{K}})$. Notice that we use the fact that an open set within an open set is open globally. ☐

Lemma 5.2.14. *In the situation of Theorem 5.2.11, for any $f \in \mathcal{A}$, any $\vec{r} \in V$ and any $P \subseteq \{1, \dots, n\}$, the function $\vec{x} \mapsto f_{P,\vec{r}_{\overline{P}}}(\vec{x}_P)$ is in $\mathcal{A}$.*

Proof. The function f is of the form

$$D_{\mathcal{S}_0}(v_0)^{-1} \circ T_{D_{\mathcal{S}_0}(u_0)} \circ A \circ \left(T_{D_{\mathcal{S}_1}(u_1)} \circ D_{\mathcal{S}_1}(v_1), \dots, T_{D_{\mathcal{S}_n}(u_n)} \circ D_{\mathcal{S}_n}(v_n)\right)$$

where $u_i, v_i \in \Sigma_{\mathcal{S}_i}^*$ for all $i = 0, \dots, n$ and A is affine: $A(\vec{x}) = \vec{a} \cdot \vec{x} + b$ (with the scalar product) for all $\vec{x} \in V$. We will conclude the proof by realizing that the function $\vec{x} \mapsto f_{P,\vec{r}_{\overline{P}}}(\vec{x}_P)$ is of the same form but with another affine function $\overline{A}$, defined as follows, instead of A:

$$\overline{A}(\vec{x}) = \vec{x}_P \cdot \vec{a}_P + \sum_{i\in\overline{P}} \left(T_{D_{\mathcal{S}_i}(u_i)}\Big(\big(D_{\mathcal{S}_i}(v_i)\big)(r_i)\Big)\right) a_i + b$$

where $(a_1, \dots, a_n) := \vec{a}$ and $(r_1, \dots, r_n) := \vec{r}$. ☐

The theorem will be proved in a way which from the intuitive perspective follows the same pattern like the proof of Proposition 4.1.13 on page 56.

Proof of Theorem 5.2.11. Like in the proof of Proposition 4.1.13, we will prove the theorem for every $f^q_{\mathcal{T}}$ instead of $f_{\mathcal{T}}$ only, by induction on the depth of q in the underlying graph $\mathcal{G}_{\mathcal{T}}$.

For *depth*$(q) = 1$, the sub-transducer consisting of states reachable from q is SC. We can apply Proposition 5.2.8 to get that either $f^q_{\mathcal{T}}$ does not have any threads or it is single-valued and not continuously differentiable on any open set or it is in the set $\mathcal{A}$. Thus any continuously differentiable thread $h\colon J \to I_{\mathcal{S}_0}$ of $f^q_{\mathcal{T}}$ is in $\mathcal{A}|_J$ and thus in the gluing closure of $\mathcal{A}|_J$.

Assume, as the induction hypothesis, that the proposition holds for all $f^{q'}_{\mathcal{T}}$ for vertices q' with *depth*$(q') < k$, i.e. that all continuously differentiable threads $h\colon J \to I_{\mathcal{S}_0}$ of $f^{q'}_{\mathcal{T}}$ on an open rectangle J, are in *Glue*$(\mathcal{A}|_J)$ if all the representations are open. Take a state q with *depth*$(q) = k$ and a continuously differentiable thread $h\colon J \to I_{\mathcal{S}_0}$ of $f^q_{\mathcal{T}}$, with $J = J_1 \times \cdots \times J_n \subseteq V$, a product of intervals open in $\mathbb{R}^n$.

We will proceed by proving that h has a constant sign of derivative and at the same time that it belongs to *Glue*$(\mathcal{A}|_J)$.

The crucial observation is that any finite computation $\gamma \in \mathcal{H}^{\mathrm{fin}}_{\mathcal{T}_q}$ leads to a state which is either of a lower depth (i.e. *depth*$(q_\gamma) < k$) or to a state from which there is a path back to q. Denote $J_\gamma := [\![\Pi_{\mathrm{In}}(\gamma)]\!]$ for every $\gamma \in \mathcal{H}^{\mathrm{fin}}_{\mathcal{T}q}$. Divide the set $\{\gamma \in \mathcal{H}^{\mathrm{fin}}_{\mathcal{T}_q} \mid J_\gamma \times [\![\Pi_0(\gamma)]\!] \subseteq \mathrm{Zone}(h)\}$ into two disjoint sets N_q and C_q according to which of these two cases, respectively, happens for γ. (N stands for i<u>n</u>duction, C for <u>c</u>ycle.)

For $\gamma \in C_q$, like in the proof of Proposition 5.2.8, we get that there is a point in J_γ, denote it by $\vec{r}_\gamma$, which is a centre of a sub-self-similarity of $f_{\mathcal{T}}$. Thus by Corollary 5.2.5, h has the spreading property in $\vec{r}_\gamma$ and by Lemma 5.2.2, $h_{P_\gamma,(\vec{r}_\gamma)_{\overline{P}_\gamma}}$ (where $P_\gamma := P^h_{\vec{r}_\gamma}$) is in *Glue*$\big((\mathcal{A}_{P_\gamma,(\vec{r}_\gamma)_{\overline{P}_\gamma}})|_J\big)$.

For $\gamma \in N_q$, we can use the induction hypothesis to derive that

$$D_{\mathcal{S}_0}(\Pi_0(\gamma))^{-1} \circ h \circ (D_{\mathcal{S}_1}(\Pi_1(\gamma)), \ldots, D_{\mathcal{S}_n}(\Pi_n(\gamma)))$$

is in *Glue*$(\mathcal{A}|_J)$ which implies that $h|_{J_\gamma}$ is in *Glue*$(\mathcal{A}|_{J_\gamma})$. (We need to prove that the same holds for J instead of J_γ.) Since by Lemma 5.2.12, the function $h|_{J_\gamma}$ has constant sign of derivative, there is a global set of 'slanted' coordinates $P_\gamma := P^{h|_{J_\gamma}}$.

We will aim at showing that also in the case $\gamma \in N_q$, h has the spreading property at some point $\vec{r}_\gamma$ in J_γ and at the same time $h_{P_\gamma,(\vec{r}_\gamma)_{\overline{P}_\gamma}}$ is in *Glue*$\big((\mathcal{A}_{P_\gamma,(\vec{r}_\gamma)_{\overline{P}_\gamma}})|_J\big)$. By Lemma 5.2.12, the former follows from the latter.

First, pick any point $\vec{x} \in (J_\gamma)^o$ and consider the following set of subsets of $J_{P_\gamma} = \{\vec{y}_{P_\gamma} \mid \vec{y} \in J\}$:

$$\left\{K \subseteq J^o_{P_\gamma} \,\middle|\, (J_\gamma)^o_{P_\gamma} \subseteq K,\ K \text{ is open and connected},\ \big(h_{P_\gamma,\vec{x}_{\overline{P}_\gamma}}\big)|_K \in \mathit{Glue}\big((\mathcal{A}_{P_\gamma,\vec{x}_{\overline{P}_\gamma}})|_K\big)\right\}. \tag{5.2}$$

If this set includes J^o_P itself, then we can put $\vec{r}_\gamma := \vec{x}$. Assume now that J^o_P is not included in the set.

By Lemma 5.2.13, the set (5.2) is closed under arbitrary unions and thus has a largest element K_M (an analogy of the interval (z, Z) defined on page 57). Since $K_M \subsetneqq J^o_P$, there is some $\vec{y} \in J$ with $\vec{y}_{\overline{P}} = \vec{x}_{\overline{P}}$, such that $\vec{y}_P \in J^o_P \setminus \overline{K_M}$ because $J^o_P \subseteq \mathbb{R}^n$ is a metric space.

Next, find a point which lies somewhere 'on the boundary' of K_M within J_P as follows. First of all, K_M is not empty—it contains $\vec{x}_P$ because $\vec{x}_P$ is in $(J_\gamma)_P$ which, in turn, is in the set (5.2). Now we have a pair $(\vec{x}_P, \vec{y}_P)$ of points one of which is within and the other outside $\overline{K_M}$. We will construct a sequence of such pairs converging to the same point in both projections. Put $\vec{x}^{(0)} := \vec{x}$ and $\vec{y}^{(0)} := \vec{y}$. For every $k \in \mathbb{N}$, define $(\vec{x}^{(k+1)}, \vec{y}^{(k+1)})$ depending on whether $\vec{z}^{(k)}_P$, where $\vec{z}^{(k)} := (\vec{x}^{(k)} + \vec{y}^{(k)})/2$, is in $\overline{K_M}$ or not. If yes, then put $(\vec{x}^{(k+1)}, \vec{y}^{(k+1)}) := (\vec{z}^{(k)}, \vec{y}^{(k)})$, if not, put $(\vec{x}^{(k+1)}, \vec{y}^{(k+1)}) := (\vec{x}^{(k)}, \vec{z}^{(k)})$. Denote the limit $\lim_{k\to\infty} \vec{x}^{(k)} = \lim_{k\to\infty} \vec{y}^{(k)}$ by $\vec{z}$. The point $\vec{z}_P$ is 'on the boundary' of K_M in the sense that each neighbourhood of $\vec{z}_P$ in J_P contains a point in $\overline{K_M}$ as well as a point outside $\overline{K_M}$.

Take an infinite computation $\delta \in \mathcal{H}^q_{\mathcal{J}}$ with $\gamma \sqsubseteq \delta$, which inputs a representation of a closed interval containing the tuple $\vec{z}$ and outputs a representation of $h(\vec{z})$.

Moreover, by Lemma 3.2.17 on page Lemma 3.2.17, we can assume that $\Pi_P(\delta)$ is an open representation of $\vec{z}_P$, i.e. that the semantics of the input of each finite prefix of δ contains $\vec{z}$ in its interior, because the representations $\mathcal{S}_1, \ldots, \mathcal{S}_n$ are open.

The computation δ has a finite prefix $\delta_0 \in \mathcal{H}^{\text{fin}}_{\mathcal{J}^q}$ (which can be assumed to fulfil $\gamma \sqsubseteq \delta_0$) such that $[\![\Pi_{\text{In}}(\delta_0)]\!] \times [\![\Pi_0(\delta_0)]\!] \subseteq \text{Zone}(h)$. Denote all the finite prefixes of δ in $\mathcal{H}^q_{\mathcal{J}}$ starting from δ_0 by δ_k for all $k \in \mathbb{N}$, so that it holds $\delta_0 \sqsubseteq \delta_1 \sqsubseteq \delta_2 \sqsubseteq \cdots \sqsubseteq \delta$. Each δ_k is either in C_q or in N_q. If any of them, say $\delta_{k'}$, would be in N_q, it would mean that $h|_{J_{\delta_{k'}}}$ would be in $\mathit{Glue}\big(\mathcal{A}|_{J_{\delta_{k'}}}\big)$. This would imply that $K_M \cup (J_{\delta_{k'}})^o_P$ would be in the set (5.2) which would be a contradiction to the maximality of K_M because $J^o_{\delta_{k'}}$ contains some $\vec{y}^{(k)} \notin \overline{K_M}$. We conclude that all δ_k are in C_q.

Now we will utilize the sub-self-similarity generated by one of the computations δ_k to find the point r_γ in which h has the spreading property. Since γ is a prefix of δ, it holds $J_\gamma \supseteq J_{\delta_k}$ for each k.

Since h is continuously differentiable, there has to be a neighbourhood Ω of $\vec{z}$, in which the partial derivatives that were not zero in $\vec{z}$, are still non-zero. Thus for any $\vec{w} \in \Omega$, $P^h_{\vec{w}} \supseteq P^h_{\vec{z}} = P$.

Take an index $k_1 \in \mathbb{N}$, such that $J_{\delta_{k_1}} \subseteq \Omega$. Since $\delta_{k_1} \in C_q$, there is δ' such that $\widehat{\delta} := \delta_{k_1}\delta' \in \mathcal{H}^{\text{fin}}_{\mathcal{J}^q}$ and $q_{\widehat{\delta}} = q$. Now define

$$\vec{r}_\gamma := \left(\text{Fix}_{D_{\mathcal{S}_1}(\Pi_1(\widehat{\delta}))}, \ldots, \text{Fix}_{D_{\mathcal{S}_n}(\Pi_n(\widehat{\delta}))}\right).$$

The point $\vec{r}_\gamma$ is in $\Omega \cap J_\gamma$ and Lemma 5.2.2 applies to it because there is a sub-self-similarity of $f^q_{\mathcal{J}}$ with centre $\vec{r}_\gamma$. This lemma gives us that $(f^q_{\mathcal{J}})_{P,(\vec{r}_\gamma)_{\overline{P}}}$ is in $\mathcal{A}_{P,(\vec{r}_\gamma)_{\overline{P}}}$. We need the following, much weaker implication of this: $h_{P,(\vec{r}_\gamma)_{\overline{P}}}$ is in $(\mathcal{A}_{P,(\vec{r}_\gamma)_{\overline{P}}})|_J$ and in particular, h has the spreading property in $\vec{r}_\gamma$ and $h_{P,(\vec{r}_\gamma)_{\overline{P}}}$ is in $\mathit{Glue}\big((\mathcal{A}_{P,(\vec{r}_\gamma)_{\overline{P}}})|_J\big)$.

We completed the proof that there is a dense set of points in J, namely the set $\{\vec{r}_\gamma \mid \gamma \in C_q \cup N_q\}$, in which h has the spreading property and at the same time the instantiation of h to these points in the coordinates with non-zero partial derivative is in the gluing closure of $\mathcal{A}|_J$. By Lemma 5.2.6, h has constant sign of derivative. Thus the value of h does not depend on the values of arguments in the coordinates $\overline{P}$, which means that $h(\vec{x}) = h_{P,(\vec{r}_\gamma)_{\overline{P}}}(\vec{x}_P)$ for an arbitrary $\gamma \in C_q \cup N_q$ and any $\vec{x} \in J$. It follows that h is in $Glue(\mathcal{A}|_J)$ by Lemma 5.2.14. $\square$

Corollary 5.2.15 (Main limitation theorem for finite non-slow IFS-transducers).
Let $\mathcal{T}$ be a precise non-ignorant finite transducer of type $(\mathcal{S}_1 \times \cdots \times \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ with all representations $\mathcal{S}_i$ being open and non-slow on the interior. Any continuously differentiable thread $h\colon J \to I_0$ of $f_\mathcal{T}$ is in $Glue(\mathcal{A}|_J)$ where $\mathcal{A}$ is the function set defined in Theorem 5.2.11.

Proof. For any open interval $L \subseteq J$ it holds $h|_L \in Glue(\mathcal{A}|_L)$ by the previous theorem. For any point $\vec{x} \in J$, there is an open interval $L_{\vec{x}} \subseteq J$ with $\vec{x} \in L_{\vec{x}}$. We can now apply Lemma 5.2.13 to get $h|_J \in Glue(\mathcal{A}|_L)$. $\square$

Notice that in since we use the gluing closure, there was no need to use the fact that a thread has a connected domain in the previous proof.

5.3 Existence of a unique stretching translator

The whole of this section has one big aim: to prove Theorem 5.1.2. We will do it in several steps. First we will prove the uniqueness of a translator indirectly, then derive a formula which a translator has to satisfy and finally show that the formula actually provides us with a translator.

5.3.1 Pre-translators, uniqueness

We will now weaken the notion of a translator to gain a better understanding of it and, in particular, to find a simpler proof of its uniqueness.

Definition 5.3.1 (Stretching pre-translator).
A continuous bijection $T\colon I \to I$ on $I \in \mathcal{I}$ is called a *stretching pre-translator* of $f \in \mathrm{dContr}_I$ if $T \circ f \circ T^{-1}$ is an affine d-contraction. Such an affine d-contraction is called a *stretched pre-translation* of f.

Again, we will usually drop the words 'stretching' and 'stretched' and call these notions simply a pre-translator and a pre-translation.

It is not the case that every d-contraction (even a quite 'nice' one) has a unique pre-translator, not even a unique pre-translation. For example, the affine d-contraction $(x+1)/2$ on $[-1,1]$ has the obvious translator $\mathrm{id}_{[-1,1]}$ as well as the translator $x \mapsto -x$ which translates it to the symmetrical contraction $(x-1)/2$. This is an instance of the following simple symmetry lemma.

Definition 5.3.2. For every $[a,b] = I \in \mathcal{I}$ denote by neg_I the bijection $x \mapsto (a+b-x) : I \to I$.

Lemma 5.3.3. *For every pre-translator* T *of a d-contraction* $f \in \mathrm{dContr}_I$ *also* $\mathrm{neg}_I \circ T$ *is a pre-translator of* f. *The pre-translations* A_1, A_2 *of* f *by* T *and* $\mathrm{neg}_I \circ T$, *respectively, are symmetrical in the following sense:* $A_1 = \mathrm{neg}_I \circ A_2 \circ \mathrm{neg}_I$.

In fact, this lemma has established a bijection between increasing and decreasing pre-translators of a d-contraction. Nevertheless, the lack of uniqueness of pre-translations and pre-translators does not end with this symmetry as is shown to the full by the following lemma.

Lemma 5.3.4. *Let* $f \in \mathrm{dContr}_I$, $A \in \mathrm{dAff}_I$ *over some* $[a,b] = I \in \mathcal{I}$ *such that* one *of the following conditions holds (for* $[\![f]\!], [\![A]\!] \subseteq [a,b]$*):*

1. $a < [\![f]\!] < b$ *and* $a < [\![A]\!] < b$
2. $a \in [\![f]\!]$, $[\![f]\!] < b$ *and* $a \in [\![A]\!]$, $[\![A]\!] < b$
3. $a < [\![f]\!]$, $b \in [\![f]\!]$ *and* $a < [\![A]\!]$, $b \in [\![A]\!]$

Then for each increasing *continuous bijection* $\widehat{T}$: $I \setminus [\![f]\!] \to I \setminus [\![A]\!]$, *there exists exactly one pre-translator* T *of* f *with* $T|_{I\setminus[\![f]\!]} = \widehat{T}$. *The pre-translation of* f *corresponding to this* T *is* A.

Proof. The formula $T \circ f \circ T^{-1} = A$ is equivalent to $T \circ f = A \circ T$ and this is equivalent to the sub-self-similarity

$$A^{-1} \circ T \circ f = T. \tag{5.3}$$

Denote $F_k := [\![f^k]\!] \setminus [\![f^{k+1}]\!]$ and $E_k := [\![A^k]\!] \setminus [\![A^{k+1}]\!]$ for all $k \in \mathbb{N}$. Notice that $f^k(F_0) = F_k$ and $A^k(E_0) = E_k$ for all $k \in \mathbb{N}$ and $\bigcup_{k=0}^{\infty} F_k = I \setminus \mathrm{Fix}_f$, $\bigcup_{k=0}^{\infty} E_k = I \setminus \mathrm{Fix}_A$.

Define T as follows. For each $k \in \mathbb{N}$ and each $x \in F_0$, define $T(f^k(x)) := A^k(\widehat{T}(x))$ (see Figure 5.1). Thus T is defined on each F_k, i.e. on the whole of $I \setminus \mathrm{Fix}_f$. Put $T(\mathrm{Fix}_f) = \mathrm{Fix}_A$. T is a bijection between each pair F_k and E_k and is therefore an auto-bijection on I. It fulfils $A^{-1} \circ T \circ f = T$ on each F_k and at Fix_f, i.e. on the whole of I. It remains to prove only that T is increasing because continuity of a bijection follows from monotonicity.

We know that T is increasing on F_0 because it coincides with $\widehat{T}$ on F_0. Since both f and A are increasing, T is increasing within each F_k. Let us now distinguish the three cases above.

1. Let $F_0 := [a,c) \cup (d,b]$ with $a < c < d < b$ and $E_0 := [a,c') \cup (d',b]$ with $a < c' < d' < b$. Since $T([a,c]) < T([d,b])$, it has to hold $T([a,c]) = [a,c']$ and $T([d,b]) = [d',b]$ by continuity and monotonicity of $\widehat{T}$.

   We will now prove by induction that T is increasing on each set $\{\mathrm{Fix}_f\} \cup \bigcup_{k=0}^{n} F_k$. It is true for $n = 0$ because $T\big([a,c)\big) < \mathrm{Fix}_A < T\big((d,b]\big)$. Assume that T is increasing on

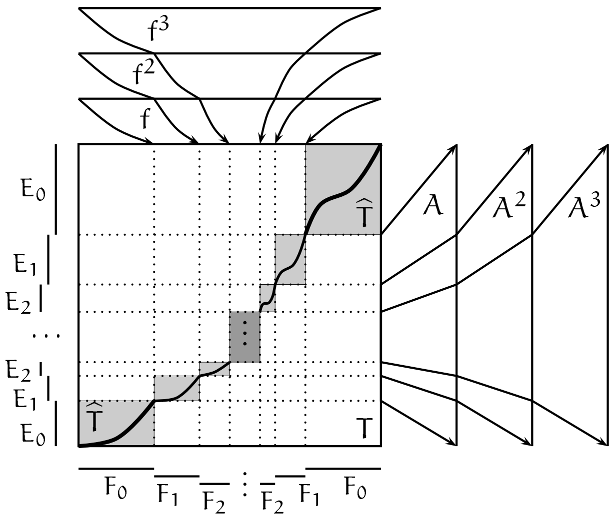

Figure 5.1: Constructing T from an arbitrary continuous and increasing $\widehat{T}$.

$\{\mathrm{Fix}_f\} \cup \bigcup_{k=0}^{n-1} F_k = [a, f^n(a)) \cup \{\mathrm{Fix}_A\} \cup (f^n(b), b]$. We already know that T is increasing on $F_n = [f^n(a), f^{n+1}(a)) \cup (f^{n+1}(b), f^n(b)]$ and that $T([f^n(a), f^{n+1}(a))) = [A^n(a), A^{n+1}(a))$ and $T((f^{n+1}(b), f^n(b)]) = (A^{n+1}(b), A^n(b)]$. This already implies that T is increasing on $[a, f^{n+1}(a)) \cup \{\mathrm{Fix}_A\} \cup (f^{n+1}(b), b] = \{\mathrm{Fix}_A\} \cup \bigcup_{k=0}^{n} F_k$.

2. Let $F_0 = [a, c)$ and $E_0 = [a, c')$, where $c, c' \in (a, b)$. Then T is increasing simply because $F_0 < F_1 < F_2 < \cdots$ and $E_0 < E_1 < E_2 < \cdots$.

3. This case is dual to 2. □

Corollary 5.3.5. *For any pair $f \in \mathrm{dContr}_I$, $A \in \mathrm{dAff}_I$ over any $I \in \mathcal{I}$, such that condition 1,2 or 3 from Lemma 5.3.4 holds, there are continuously many pre-translators of f with pre-translation A.*

This ambiguity can be eliminated by a requirement on the pre-translator to be differentiable in the fixpoint of the associated d-contraction as follows.

Proposition 5.3.6 (Unique pre-translator with derivative).
Let f be a d-contraction on some $I \in \mathcal{I}$, and let T_1, T_2 be pre-translators of f such that both derivatives $T_1'(\mathrm{Fix}_f)$ and $T_2'(\mathrm{Fix}_f)$ exist and are different from 0, $+\infty$ and $-\infty$. Then it holds either $T_1 = T_2$ or $T_1 = \mathrm{neg}_I \circ T_2$.

Proof. Let us give names to the two affine pre-translations of f:

$$A := T_1 \circ f \circ T_1^{-1} \qquad B := T_2 \circ f \circ T_2^{-1}.$$

Notice that the fixpoints of A and B are $T_1(\mathrm{Fix}_f)$ and $T_2(\mathrm{Fix}_f)$, respectively. The two equalities above allow to express f in two different ways:

$$f = T_1^{-1} \circ A \circ T_1 = T_2^{-1} \circ B \circ T_2$$

which yields an affine sub-self-similarity of $T_1 \circ T_2^{-1}$:

$$T_1 \circ T_2^{-1} = B^{-1} \circ T_1 \circ T_2^{-1} \circ A. \tag{5.4}$$

As we assume that T_1 and T_2 are differentiable in Fix_f, the function $T_1 \circ T_2^{-1}$ (mapping $T_2(\mathrm{Fix}_f)$ to $T_1(\mathrm{Fix}_f)$) is differentiable in $T_2(\mathrm{Fix}_f)$. Moreover, the derivative

$$(T_1 \circ T_2^{-1})'(T_2(\mathrm{Fix}_f)) = \frac{T_1'\left(T_2^{-1}(T_2(\mathrm{Fix}_f))\right)}{T_2'\left(T_2^{-1}(T_2(\mathrm{Fix}_f))\right)} = \frac{T_1'(\mathrm{Fix}_f)}{T_2'(\mathrm{Fix}_f)}$$

is not 0 nor infinite by the assumptions of the proposition.

Thus we get by Corollary 4.1.7 applied on (5.4), that $T_1 \circ T_2^{-1}$ is an affine bijection of I. There are only two affine bijections of I, namely id_I and neg_I. Therefore either $T_1 = T_2$ or $T_1 = \mathrm{neg}_I \circ T_2$. □

Corollary 5.3.7. *Every d-contraction has at most one translator.*

Proof. A translator of $f \in \mathrm{dContr}_I$ is an increasing pre-translator of f which is continuously differentiable, and its derivative is nowhere 0 nor $+\infty$. Thus it satisfies the conditions on T_1 and T_2 in Proposition 5.3.6. This proves that there are at most two different translators of f and if there are two, say T_1 and T_2, it holds $T_2 = \mathrm{neg}_I \circ T_1$. Since one of T_1, T_2 is increasing and the other decreasing, it is impossible that both T and $\mathrm{neg}_I \circ T$ would be translators because a translator has to be increasing. Therefore there is at most one translator of f. □

5.3.2 A limit formula for a translator

Although we have already proved uniqueness of a translator, it will be useful to give another proof because it will lead us to a formula for the translator. The formula will later become the basis for the proof of existence of a translator.

Lemma 5.3.8 (Translator satisfies a limit formula).
For a translator T of a d-contraction f on I, the following limit converges and the equation holds:

$$\lim_{n\to\infty} \frac{(f^n)'(x)}{(f'(\mathrm{Fix}_f))^n} = \frac{T'(x)}{T'(\mathrm{Fix}_f)} \tag{5.5}$$

for every $x \in I$. Moreover, $T'(x)$ is not equal to 0 nor $+\infty$ for any $x \in I$.

Proof. Since T and f are differentiable, we can use the formula for the derivative of a composition:

$$(T \circ f \circ T^{-1})' = \frac{(T' \circ f \circ T^{-1}) \cdot (f' \circ T^{-1})}{T' \circ T^{-1}} = \frac{(T' \circ f) \cdot f'}{T'} \circ T^{-1} \tag{5.6}$$

As this derivative is constant and T is a bijection, we get that also

$$\frac{\big(T'(f(x))\big) \cdot f'(x)}{T'(x)} \tag{5.7}$$

is constant for all x where $T'(x)$ and $T'(f(x))$ are not in $\{0, +\infty\}$.

Next, we will show that for each $x \in I$, $T'(x)$ is not in $\{0, +\infty\}$. With $A := T \circ f \circ T^{-1}$, the sub-self-similarity $A^{-1} \circ T \circ f = T$ holds (the same as equation (5.3) on page 90). Assuming that $T'(x) \in \{0, +\infty\}$, this sub-self-similarity implies that also $T'(f(x)), T'(f^2(x)), \ldots$ are in $\{0, +\infty\}$. By continuity of T' it follows that

$$T'(\mathrm{Fix}_f) = T'\left(\lim_{k\to\infty} f^k(x)\right) = \lim_{k\to\infty} T'(f^k(x)) \in \{0, +\infty\}$$

which is a contradiction to the definition of a translator.

Substituting Fix_f for x in equation (5.7) reveals that the expression is equal to $f'(\mathrm{Fix}_f)$, i.e. Strg_f. Therefore

$$f'(x) \cdot \frac{T'(f(x))}{T'(x)} = f'(\mathrm{Fix}_f) \quad \text{which is equivalent to} \quad \frac{f'(x)}{f'(\mathrm{Fix}_f)} = \frac{T'(x)}{T'(f(x))} \tag{5.8}$$

holds for all $x \in I$.

Formula (5.8) can be generalized by substituting f^n for f:

$$\frac{(f^n)'(x)}{(f^n)'(\mathrm{Fix}_f)} = \frac{T'(x)}{T'(f^n(x))} \tag{5.9}$$

because T is a translator for the d-contraction f^k as well as for f.

Using the continuity of T' and the equation $(f^n)'(\mathrm{Fix}_f) = (f'(\mathrm{Fix}_f))^n$, which is an easy consequence of the chain rule, we get equality (5.5) as follows:

$$\lim_{n\to\infty} \frac{(f^n)'(x)}{(f'(\mathrm{Fix}_f))^n} = \lim_{n\to\infty} \frac{(f^n)'(x)}{(f^n)'(\mathrm{Fix}_f)} = \lim_{n\to\infty} \frac{T'(x)}{T'(f^n(x))} = \frac{T'(x)}{T'(\mathrm{Fix}_f)} \qquad \square$$

Having proved equation (5.5), it is a simple matter to derive a formula for a translator T of f and check that it really defines a translator as long as it converges in a nice enough manner. This is the content of the next proposition.

Proposition 5.3.9. *Let f be a d-contraction on $I = [a, b] \in \mathcal{I}$ such that the limit*

$$g(x) := \lim_{n\to\infty} \frac{(f^n)'(x)}{(f'(\mathrm{Fix}_f))^n} \tag{5.10}$$

converges for each $x \in I$ *to a positive real and* g *is continuous on* I*. Then* f *has a translator* T *defined as follows*

$$T(x) = a + (b-a)\frac{\int_a^x g(y)dy}{\int_a^b g(y)dy}. \tag{5.11}$$

Proof. The definition of T is correct because as a continuous function, g is integrable. Since g is positive, the integral function $\int_a^x g(y)dy$ is increasing. Therefore also T is increasing. It is easy to check that $T(a) = a$ and $T(b) = b$, which is sufficient to conclude that T is an increasing bijection of I.

We will prove that $T \circ f \circ T^{-1}$ is affine by calculating its derivative and establishing that it is constant. By the definition of T, we have

$$\frac{\int_a^b g(y)dy}{b-a} \cdot T'(x) = g(x) =$$

$$\lim_{n\to\infty} \frac{(f^n)'(x)}{(f'(\mathrm{Fix}_f))^n} = \lim_{n\to\infty} \frac{(f^{n-1})'(f(x)) \cdot f'(x)}{(f'(\mathrm{Fix}_f))^n} = \frac{f'(x)}{f'(\mathrm{Fix}_f)} \cdot \lim_{n\to\infty} \frac{(f^{n-1})'(f(x))}{(f'(\mathrm{Fix}_f))^{n-1}} =$$

$$\frac{f'(x)}{f'(\mathrm{Fix}_f)} \cdot g(f(x)) = \frac{f'(x)}{f'(\mathrm{Fix}_f)} \cdot T'(f(x)) \cdot \frac{\int_a^b g(y)dy}{b-a}$$

for all $x \in I$. By substituting $T^{-1}(x)$ for x in the equation and by cancellation of common factors, we get

$$T'(f(T^{-1}(x))) \cdot \frac{f'(T^{-1}(x))}{f'(\mathrm{Fix}_f)} = T'(T^{-1}(x))$$

for all $x \in I$, which results in

$$(T \circ f \circ T^{-1})'(x) = \frac{T'(f(T^{-1}(x))) \cdot f'(T^{-1}(x))}{T'(T^{-1}(x))} = f'(\mathrm{Fix}_f)$$

for all $x \in I$. □

5.3.3 Convergence of the formula

As a final ingredient for the proof of Theorem 5.1.2, we need to prove that the limit (5.10) converges uniformly in x for any non-slow d-contraction f. Before we will be able to do that, we need to state the following standard property of infinite products: (see e.g. [Rudin 1986, Theorem 15.4]):

Proposition 5.3.10. *The infinite product* $\prod_{n=0}^{\infty}(1 + t_n(x))$*, where* $t_n\colon I \to (-1, \infty)$ *are continuous functions from some interval* $I \in \mathcal{I}$ *with* $\mathrm{sign}(t_n(x)) = \mathrm{sign}(t_m(x))$ *for every* $n, m \in \mathbb{N}$ *and* $x \in I$*, uniformly converges to some* $t\colon I \to (0, \infty)$ *if and only if the infinite sum* $\sum_{n=0}^{\infty} t_n(x)$ *converges uniformly.*

Proposition 5.3.11 (Convergence).
For any $f \in \mathrm{dContr}_I$ *which is not slow, the limit (5.10) converges to a positive real number for every* $x \in I$ *and the function* g *is continuous in* x.

Proof. Denote $F := \mathrm{Fix}_f$, $w := f'(F) = \mathrm{Strg}_f$, $C := f''(F)$ and let all expressions of the form $[A \pm B]$ denote the interval $[A - B, A + B]$, etc.

- The limit (5.10) is a disguised infinite product:

$$\lim_{n\to\infty} \frac{(f^n)'(x)}{w^n} = \lim_{n\to\infty} \frac{f'(f^{n-1}(x))}{w} \cdot \frac{f'(f^{n-2}(x))}{w} \cdot \dots \cdot \frac{f'(f(x))}{w} \cdot \frac{f'(x)}{w} = \prod_{n=0}^{\infty} \frac{f'(f^n(x))}{w} = \prod_{n=0}^{\infty} \left(1 + \frac{f'(f^n(x)) - w}{w}\right).$$

  Denote the partial product of the first k factors by $p_k(x)$. Each p_k is continuous in x.

- In order to show the convergence by Proposition 5.3.10, we need bounds on the expression $f'(f^n(x)) - w$. These can be derived from the formulas for the first and second derivatives of f in F:

$$w = f'(F) = \lim_{x\to F} \frac{f(x) - F}{x - F} \quad \text{and} \quad C = f''(F) = \lim_{x\to F} \frac{f'(x) - w}{x - F}. \tag{5.12}$$

  The former one implies that for each $\delta > 0$ there is $\xi_1(\delta) > 0$ with $f(x) - F \in (x - F) \cdot [w \pm \delta]$ for all $x \in [F \pm \xi_1(\delta)]$. Since f is a d-contraction, we can iterate the formula and obtain $f^n(x) - F \in (x - F) \cdot [(w \pm \delta)^n]$.

  The latter equation in (5.12) implies that for each $\delta > 0$, there is $\xi_2(\delta) > 0$ with $f'(x) - w \in (x - F) \cdot [C \pm \delta]$ for all $x \in [F \pm \xi_2(\delta)]$. Combining these two formulas for $x \in [F \pm \xi(\delta)]$ where $\xi(\delta) = \min(\xi_1(\delta), \xi_2(\delta)]$, we get:

$$f'(f^n(x)) - w \in (f^n(x) - F) \cdot [C \pm \delta] \subset [C \pm \delta] \cdot [(w \pm \delta)^n] \cdot (x - F) \tag{5.13}$$

- We will show that $\lim_{n\to\infty} p_n(x)$ uniformly converges by using the following inequalities:

$$p_{k(\delta)}(x) \cdot \prod_{i=k(\delta)}^{\infty} \left(1 + \frac{(C - \delta)(w - \delta)^i (x - F)}{w}\right) \leqslant \liminf_{n\to\infty} p_n(x) \leqslant \limsup_{n\to\infty} p_n(x) \leqslant p_{k(\delta)}(x) \cdot \prod_{i=k(\delta)}^{\infty} \left(1 + \frac{(C + \delta)(w + \delta)^i (x - F)}{w}\right) \tag{5.14}$$

  where $k(\delta)$ is the least k with $\bar{f}^k(I) \subset [F \pm \xi(\delta)]$. This inequality holds for all sufficiently small $\delta > 0$ and all $x \in I$, $x \geqslant F$ when $C > 0$ and for all $x \leqslant F$ when $C < 0$ because of

(5.13). The following dual equality holds for $x \leqslant F$ when $C > 0$ and for $x \geqslant F$ when $C < 0$:

$$p_{k(\delta)}(x) \cdot h_{k(\delta)}(\delta, x) \leqslant \liminf_{n\to\infty} p_n(x) \leqslant \limsup_{n\to\infty} p_n(x) \leqslant p_{k(\delta)}(x) \cdot h_{k(\delta)}(-\delta, x) \tag{5.15}$$

$$\text{where} \quad h_k(\delta, x) := \prod_{i=k}^{\infty} \left(1 + \frac{(C+\delta)(w+\delta)^i (x-F)}{w}\right).$$

All that we need to show is that

$$\lim_{\delta\to 0}\Big(p_{k(\delta)}(x) \cdot \big(h_{k(\delta)}(\delta, x) - h_{k(\delta)}(-\delta, x)\big)\Big) \tag{5.16}$$

converges to the constantly zero function uniformly in x. Then it will follow from the inequalities (5.14) and (5.15) that $\lim_{\delta\to 0} p_{k(\delta)}(x) \cdot h_{k(\delta)}(\delta, x)$ is a continuous function (the sequence $p_{k(\frac{1}{m})}(x) \cdot h_{k(\frac{1}{m})}(\frac{1}{m}, x)$ for $m \in \mathbb{N}$, of continuous functions uniformly converges to it) and that both functions $\liminf_{n\to\infty} p_n(x)$ and $\limsup_{n\to\infty} p_n(x)$ are equal to it.

– Firstly, we will show that there is a uniform bound on $p_k(x)$, i.e. a number $M > 0$ with $0 < p_k(x) < M$ for all $k \in \mathbb{N}$ and $x \in I$. Fix some $\delta > 0$ with $w + \delta < 1$. For any $x \geqslant F$ it follows from (5.13) that

$$p_k(x) = \prod_{n=0}^{k} \left(1 + \frac{f'(f^n(x)) - w}{w}\right) \leqslant p_{k(\delta)}(x) \cdot \prod_{n=k(\delta)}^{\infty} \left(1 + \frac{(C+\delta)(w+\delta)^n (x-F)}{w}\right) \leqslant$$
$$\sup_{x\in I}\big(p_{k(\delta)}(x)\big) \cdot \prod_{n=k(\delta)}^{\infty} \left(1 + \frac{(C+\delta)(w+\delta)^n \cdot |I|}{w}\right) \tag{5.17}$$

The right hand side of the inequality is a positive real constant because a continuous function, i.e. $p_{k(\delta)}(x)$, over a compact interval is bounded and because by Proposition 5.3.10 and the following equation, the infinite product converges.

$$\sum_{n=k(\delta)}^{\infty} \left(\frac{(C+\delta)(w+\delta)^n \cdot |I|}{w}\right) \leqslant \frac{(C+\delta) \cdot |I|}{w} \cdot \frac{1}{1-w-\delta}$$

For $x < F$, we can show a uniform bound analogously. Take M to be the larger one of the two bounds.

– Since there is a uniform bound on $p_k(x)$, we can prove that (5.16) converges to zero uniformly in x, by proving that $\lim_{\delta\to 0}\big(h_k(\delta, x) - h_k(-\delta, x)\big)$ converges to zero uniformly in x for any $k \in \mathbb{N}$. This follows from the fact that every h_k is continuous in both δ and x (see below) and that the domain I of x is closed. The function h_k is continuous by Proposition 5.3.10 and the fact that the following infinite sum converges uniformly in both x and δ:

$$\sum_{n=k}^{\infty} \left(\frac{(C+\delta)(w+\delta)^n (x-F)}{w}\right) = \frac{(C+\delta)(x-F)}{w} \cdot \frac{1-(w-\delta)^k}{1-w-\delta} \qquad \square$$

Proof of Theorem 5.1.2 on page 76. We need to prove that a non-slow d-contraction has a unique translator. The existence of the translator follows from Propositions 5.3.9 and 5.3.11 and the uniqueness from Corollary 5.3.7. The property $T'(x) \notin \{0, +\infty\}$ has been proved as a part of Lemma 5.3.8. $\square$

Chapter 6

Finite Automata and LFT-Representations

LFT stands for *linear fractional transformation* also called *Möbius transformation*. These are names referring to a function given by a formula of the following form:

$$x \mapsto \frac{ax + c}{bx + d}$$

for some a, b, c, d, usually defined over the field of real or complex numbers extended with ∞.

Complex LFT's play an important role in complex analysis while the real LFT's have been used as digit functions in IFS-representations of real numbers [Edalat & Potts 1997, Potts 1998]. Their advantage over affine digit functions is that they generalize continued fraction representations and allow one to use results from continued fraction theory to implement elegantly many useful functions with LFT-representations (see [Potts 1998]).

6.1 Linear fractional transformations

We will start by drawing a view of real LFT's as arising from a part of linear algebra. This insight will shed light on some of the crucial properties of real LFT's.

6.1.1 Projective factorization of linear maps

Recall that we used the extended real numbers, $\mathbb{R}^*$, to define the derivative of a many-valued function. These numbers were used to represent the *direction(s)* of curves within the graph of a many-valued function f when passing through a particular point.

The notion of direction in a vector space is formalized in linear algebra as an element of its projective space. For example, the set $\mathbb{R}^*$ is isomorphic to the projective space over $\mathbb{R}^2 = \mathbb{R} \times \mathbb{R}$ which is the factor of $\mathbb{R}^2 \setminus \{(0,0)\}$ by the equivalence relation $\sim$ defined by

$$(x, y) \sim (x', y') \iff (\exists t \in \mathbb{R})(xt = x',\ yt = y').$$

This relation equates any two points in $\mathbb{R}^2 \setminus \{(0,0)\}$ if and only if they are colinear with the origin, i.e. if and only if they lie in one and the same direction or in directly opposite directions from the origin.

The isomorphism between $\mathbb{R}^2 \setminus \{(0,0)\}/\sim$ and $\mathbb{R}^*$ relates each class $[(x,y)]_\sim$ with the number x/y.

It is also possible to factor a linear mapping $L\colon \mathbb{R}^2 \to \mathbb{R}^2$ by the same equivalence relation provided that $L^{-1}(0,0) = \{(0,0)\}$. The linear mapping $L(x,y) = (ax + cy, bx + dy)$ can be represented by a $(2,2)$ matrix $\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)$ in the sense that it holds:

$$\big(L(x,y)\big)^\top = \begin{pmatrix} a & c \\ b & d \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$$

for any $x, y \in \mathbb{R}$.

It can be checked that the condition $L^{-1}(0,0) = \{(0,0)\}$ holds if and only if the matrix $\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)$ is regular.

The promised factor of L by $\sim$ is the following LFT:

$$L/\!\sim\; = \left(\frac{x}{y} \mapsto \frac{ax + cy}{bx + dy}\right) = \left(\frac{x}{y} \mapsto \frac{a\frac{x}{y} + c}{b\frac{x}{y} + d}\right) = \left(x \mapsto \frac{ax + c}{bx + d}\right) : \mathbb{R}^* \to \mathbb{R}^*.$$

Clearly, any LFT arises as a factor $L/\!\sim$ of some linear map $L\colon \mathbb{R}^2 \to \mathbb{R}^2$. Two linear maps $\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)$ and $\left(\begin{smallmatrix} a' & c' \\ b' & d' \end{smallmatrix}\right)$ correspond to the same LFT if and only if $\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right) = k \cdot \left(\begin{smallmatrix} a' & c' \\ b' & d' \end{smallmatrix}\right)$ for some $k \in \mathbb{R} \setminus \{0\}$.

To denote a particular LFT L in the rest of this section, we will often use the matrix that represents one of the regular linear maps $\mathbb{R}^2 \to \mathbb{R}^2$ from the class which corresponds to L. There is no risk of confusion since we will never need to use matrices to denote anything else but LFT's.

By looking at the underlying linear maps, it is easy to see that a composition of two LFT's $\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)$ and $\left(\begin{smallmatrix} a' & c' \\ b' & d' \end{smallmatrix}\right)$ is again an LFT and is equal to the matrix product $\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right) \cdot \left(\begin{smallmatrix} a' & c' \\ b' & d' \end{smallmatrix}\right)$.

The parallel between classes of matrices and LFT's implies the following proposition:

Proposition 6.1.1. *The LFT's with binary composition and unit* $\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ *form a group, denoted* LFT, *which is isomorphic to* SL(2).

Every LFT is a continuous isomorphism of $\mathbb{R}^*$. *The LFT's* $\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)$ *and* $\left(\begin{smallmatrix} d & -c \\ -b & a \end{smallmatrix}\right)$ *are inverses of each other.*

The affine maps $\mathbb{R} \to \mathbb{R}$ *extended to* $\mathbb{R}^*$ *by mapping* ∞ *to itself, form a subgroup of* LFT *which consists of LFT's of the form* $\left(\begin{smallmatrix} a & c \\ 0 & d \end{smallmatrix}\right)$.

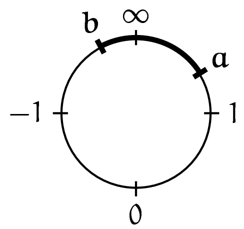

Figure 6.1: An extended interval $[a, b]$ in $\mathbb{R}^*$.

The proof of the following famous property can be found in many introductory analysis books, e.g. in [Rudin 1986, Section 14.3].

Lemma 6.1.2 (Three point characterization of LFT).
Let $x_1, y_1, z_1, x_2, y_2, z_2 \in \mathbb{R}^*$ such that $x_1 \neq y_1 \neq z_1 \neq x_1$ and $x_2 \neq y_2 \neq z_2 \neq x_2$. There is one and only one LFT L such that $L(x_1) = x_2$, $L(y_1) = y_2$ and $L(z_1) = z_2$.

Corollary 6.1.3. *The function set* LFT *is locally determinable.*

Proof. In any open set $U \subseteq \mathbb{R}^*$ there exist three distinct points. By knowing the values of an LFT in these three points, the LFT is fully determined. □

6.1.2 Extended intervals and LFT's

A similar definition to the following one is used by Edalat & Potts [1997].

Definition 6.1.4. A closed *extended interval* in $\mathbb{R}^*$ is a proper subset of $\mathbb{R}^*$ which is connected and closed. It can be shown that each closed extended interval is in the following form:

$$[a, b] := \begin{cases} \{x \in \mathbb{R} \mid a \leqslant x \leqslant b\} & \text{for } a, b \in \mathbb{R},\ a < b, \\ \{\infty\} \cup \{x \in \mathbb{R} \mid x \geqslant a\} \cup \{x \in \mathbb{R} \mid x \leqslant b\} & \text{for } a, b \in \mathbb{R},\ a > b, \\ \{\infty\} \cup \{x \in \mathbb{R} \mid x \geqslant a\} & \text{for } a \in \mathbb{R},\ b = \infty, \\ \{\infty\} \cup \{x \in \mathbb{R} \mid x \leqslant b\} & \text{for } a = \infty,\ b \in \mathbb{R}, \\ \{a\} & \text{for } a = b \in \mathbb{R}^*. \end{cases}$$

Every non-singleton extended interval $[a, b]$ can be visualized as a section of the circle representing $\mathbb{R}^*$ with endpoints a and b, going from a to b in the positive sense (see Figure 6.1).

Denote the set of all closed *non-singleton* extended intervals by $\mathcal{I}^*$.

The following lemma is another corollary of Lemma 6.1.2 above.

Lemma 6.1.5. *For each pair $I_1, I_2 \in \mathcal{I}^*$ there is an LFT L which maps I_1 onto I_2.*

The notion of IFS-representation can be generalized to extended intervals using some LFT translation of the extended interval to an ordinary interval as follows:

Definition 6.1.6. An *extended real number IFS-representation* is a tuple (I, L, Σ, D) where $I \in \mathcal{I}^*$ and L is an LFT such that $L(I) \in \mathcal{I}$ and $(L(I), \Sigma, \mathfrak{d} \mapsto L \circ D(\mathfrak{d}) \circ L^{-1})$ is an IFS-representation.

Let us consider all of the notions derived from IFS-representation (e.g. IFS-transducer, semantics of digit sequences) generalized also for extended IFS-representations in the straightforward way.

6.1.3 LFT's of multiple-arity

For the formulation of the limitation theorem for transducers with multiple input channels, we need the notion of n-ary LFT.

An n-ary LFT is a multi-linear map $L\colon \left(\mathbb{R}^2\right)^n \to \mathbb{R}^2$ which fulfils

$$L^{-1}\big((0,0)\big) = \big\{\big((0,0), \ldots, (0,0)\big)\big\},$$

factored by the projective equivalence $\sim$.

The LFT's which we considered so far were all unary. Unless stated otherwise, by an LFT without a specified arity we mean a unary LFT.

Extended real numbers coincide with nullary LFT's.

Let us treat an n-ary map $L\colon (\mathbb{R}^*)^n \to \mathbb{R}^*$ whose value does not depend on the arguments in coordinates $\overline{P}$ and whose instantiation L_P is an $|P|$-ary LFT, as a $|P|$-ary LFT. Let us use the phrase $\leqslant$*n-ary LFT* to denote an LFT of arity less or equal to n. Thus every affine map $A\colon \mathbb{R}^n \to \mathbb{R}$, extended to $\mathbb{R}^*$ as usual, is a $\left|P^A\right|$-ary LFT, i.e. a $\leqslant n$-ary LFT.

Multi-linear maps $L\colon \left(\mathbb{R}^2\right)^n \to \mathbb{R}^2$ can be characterized by regular $(n+1)$-dimensional tensors of type $(\underbrace{2, \ldots, 2}_{(n+1)\times})$ and every composition of such linear maps corresponds to a certain tensor product. Therefore also LFT's of higher arity are closed on arbitrary finite composition.

The following lemma states this for a special case of such a composition:

Lemma 6.1.7. *For any affine function $A(\vec{x}) = \vec{a} \cdot \vec{x} + b$ and LFT's $L_0, \ldots, L_n\colon \mathbb{R}^* \to \mathbb{R}^*$, the following composition is an $\left|P^A\right|$-ary LFT:*

$$L_0 \circ A \circ (L_1, \ldots, L_n)$$

Definition 6.1.8. Let aLFT^n denote the set of all $\leqslant n$-ary LFT's of the form from the previous lemma.

Proposition 6.1.9. *The function set of all $\leqslant n$-ary LFT's of type $(\mathbb{R}^*)^n \to \mathbb{R}^*$ is locally determinable.*

Proof. First note that the set of $\leqslant 1$-ary LFT's is locally determinable by Corollary 6.1.3 and the fact that constant functions are locally determinable and also locally distinguishable from unary LFT's.

Take any open set $U \subseteq (\mathbb{R}^*)^n$, any $\leqslant n$-ary LFT L and any $\vec{x} \in (\mathbb{R}^*)^n \setminus U$. We will show that $L(\vec{x})$ can be determined from the values of L on U.

Consider the line N in $(\mathbb{R}^*)^n$ which connects some point in U with $\vec{x}$. The intersection $N \cap U$ is open within N.

Let $\vec{\xi}\colon \mathbb{R}^* \to (\mathbb{R}^*)^n$ be some affine parametrization of N with $\xi(0) = \vec{x}$. The composition $L \circ \vec{\xi}$ is a $\leqslant 1$-ary LFT and therefore it is determined by its values on $N \cap U$. Therefore $L(\vec{x}) = L(\xi(0))$ is determined by the values of L on $N \cap U \subset U$. $\square$

6.2 LFT's as d-contractions

When we study LFT's as d-contractions, we need to know something about their fixpoints. Below, there are those standard definitions and statements in this area which will be useful to us. We do not list all of the related basic properties of LFT's nor study under what conditions a set of LFT's can make an IFS-representation. Instead, we take the fastest route to the limitation theorem for LFT-representations. For a wider study of LFT's as contractions see e.g. [Potts 1998, Heckmann 1998*b*].

Lemma 6.2.1. *Every LFT has at most 2 fixpoints.*

Proof. Fixpoints of an LFT correspond to the direction of the eigenvectors of any of the underlying linear maps. Therefore there are at most two of them. $\square$

Definition 6.2.2 (Attracting and repelling fixpoints).
A fixpoint $x \in \mathbb{R}$ of a function $f\colon \mathbb{R} \to \mathbb{R}^*$ is attracting on the left, respective on the right, if there exists some $\varepsilon > 0$ such that for any $z \in [x - \varepsilon, x)$, respective $z \in (x, x + \varepsilon]$, it holds $|f(z) - x| < |z - x|$.

A fixpoint which is attracting, respective repelling, on both sides is called attracting, respective repelling.

We will state without a proof the following well-known properties of fixpoints of real functions:

Lemma 6.2.3. *In the situation of preceding definition, if $f'(x)$ exists and $|f'(x)| < 1$, then x is attracting. If $|f'(x)| > 1$ then x is repelling.*

Moreover, if $f'(x) = 1$ and $f''(x)$ exists and $f''(x) > 0$ then x is attracting on the left and repelling on the right. If $f'(x) = 1$ and $f''(x) < 0$, then x is repelling on the left and attracting on the right.

Corollary 6.2.4. *Let* $r \in \mathbb{R}$ *be a fixpoint of an LFT* L, $L \neq \mathrm{id}_{\mathbb{R}^*}$. *If* $L'(r) = 1$, *then* r *is attracting from one side and repelling from the other.*

Proof. Assume that $L = \left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)$. Calculate a formula for the first and second derivative of L:

$$L'(x) = \left(\frac{ax+c}{bx+d}\right)' = \frac{ad-bc}{(bx+d)^2} \quad \text{and} \quad L''(x) = \frac{2b(ad-bc)\cdot(bx+d)}{(bx+d)^4}.$$

Since $L'(r) = 1$, it holds $(br+d) \neq 0$. Therefore if $b \neq 0$, then $L''(r) \neq 0$ and we can apply the previous lemma to get the result. If $b = 0$, from $L'(r) = 1$ we get $a = d \neq 0$. This means $L(x) = x + c/d$ which is either the identity or it does not have any fixpoint—a contradiction. ☐

The following lemma characterizes all LFT's that have exactly two fixpoints one of which is finite and attracting.

Lemma 6.2.5. *For each triple* (r, s, t) *where* $r \in \mathbb{R}$, $s \in \mathbb{R}^*$, $r \neq s$, *and* $t \in (0, 1)$, *there is exactly one LFT* $L_{r,s,t}$ *with fixpoints* r *and* s, *such that the derivative* $L'_{r,s,t}(r)$ *is equal to* t. *The LFT* $L_{r,s,t}$ *is given by one of the following formulas:*

$$L_{r,s,t} = \begin{pmatrix} ts-r & (1-t)rs \\ t-1 & s-tr \end{pmatrix} \qquad \textit{for } s \neq \infty \tag{6.1}$$

$$L_{r,\infty,t} = \begin{pmatrix} t & (1-t)r \\ 0 & 1 \end{pmatrix}. \tag{6.2}$$

Proof. Let $L = \left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)$ be an LFT with fixpoints r, s and with $L'(r) = t$. Assuming first that $s \neq \infty$, the following equations hold:

$$ar + c = br^2 + dr, \qquad as + c = bs^2 + ds, \qquad ad - bc = t(br+d)^2. \tag{6.3}$$

We will show that this system implies the following equations linear in a, b, c, d:

$$a = b(r+s) + d, \qquad -c = b \cdot rs, \qquad a - br = t(a - bs). \tag{6.4}$$

The first two equations in (6.4) follow from the first two equations of (6.3) because the former ones are linear combinations of the latter ones. The third equation in (6.4) follows from the third equation in (6.3) and the first two in (6.4) as follows:

$$\underline{(a-br)(a-bs)} = a^2 - ab(r+s) + b^2rs = a(a - b(r+s)) + b^2rs =$$
$$ad - bc = t(br+d)^2 = t(br + a - b(r+s))^2 = \underline{t(a-bs)^2}$$

It holds $a - bs \neq 0$ because $ad - bc \neq 0$. Therefore we can cancel the common factor $a - bs$ in the above equation.

The homogeneous linear equations in (6.4) are clearly linearly independent and therefore their solution is unique up to a multiplication by a constant. This proves the uniqueness of $L_{r,s,t}$ and at the same time helps us to derive the formula (6.1) for $L_{r,s,t}$. It can be easily checked that it satisfies all of the criteria and thus proves the existence.

Assume now that $s = \infty$. This implies, first of all, that $b = 0$ and this further implies

$$ar + c = dr, \qquad a = dt$$

which, again, is a linearly independent homogeneous system of linear equations yielding a unique solution up to a multiplication by a constant, which corresponds to the formula (6.2). □

Now we will apply these lemmas about LFT's on LFT's which can be used as d-contractions on some interval I. Since we haven't formally defined LFT d-contractions and LFT-representations yet, let us do it now.

Definition 6.2.6 (LFT d-contractions on I).
For an interval $I \in \mathcal{I}$, denote by dLFT_I the set of all d-contractions that are equal to $L|_I$ for some LFT L.

Definition 6.2.7 (LFT-representation).
An IFS-representation (I, Σ, D) in which $D(\Sigma) \subset \mathrm{dLFT}_I$, is called an LFT-representation.

An extended IFS-representation (I, L, Σ, D) in which every $D(\mathfrak{d})$ for $\mathfrak{d} \in \Sigma$, is an LFT, is called an extended LFT-representation.

Computation with an extended LFT-representation (I, L, Σ, D) amounts to computing with the ordinary LFT-representation $(L(I), \Sigma, \mathfrak{d} \mapsto L \circ D(\mathfrak{d}) \circ L^{-1})$ in which both input and output are translated by L. This will allow us to prove the limitation theorem for extended LFT-representations easily as a corollary of its version for ordinary LFT-representations.

Lemma 6.2.8. *A non-slow $f \in \mathrm{dLFT}_I$ is equal to some $L_{r,s,t}$.*

Proof. Since $f'(\mathrm{Fix}_f) < 1$, we can put $r := \mathrm{Fix}_f$ and $t := f'(\mathrm{Fix}_f)$. It remains to be proved that f has another fixpoint $s \in \mathbb{R}^*$. Assume, on the contrary, that r is the only fixpoint of f in $\mathbb{R}^*$. The function $g(x) := f(x) - x$ has only one root in $\mathbb{R}$, i.e. r. Moreover, $g(x) > 0$ in some left neighbourhood of r and $g(x) < 0$ in some right neighbourhood of r because f is a d-contraction and r is attracting. By the continuity of f on $\mathbb{R}$, this implies that $g(x)$ is positive for all $x < r$ and negative for all $x > r$. By the continuity of f on $\mathbb{R}^*$, we get $f(\infty) = \infty$, a second fixpoint, which is a contradiction. □

Proposition 6.2.9. *Every LFT-representation is non-slow on the interior.*

Proof. Let (I, Σ, D) be an LFT-representation. Assume that for some $u \in \Sigma^*$, the LFT d-contraction $D(u)$ is slow. By Corollary 6.2.4, the fixpoint of $D(u)$ has to be an endpoint because otherwise it would be repelling within I on one side. □

6.3 The limitation theorem

We will be able to prove the main limitation theorem for LFT-representations by using Corollary 5.2.15 as soon as we figure out what is the translator of an arbitrary LFT d-contraction. The trick is to guess the translator by assuming that it might be again an LFT. Although it is possible and even exciting to derive the translator by using the formula in Proposition 5.3.9 on page 93, it is not necessary at all. The only benefit of this tedious process would be perhaps gaining more confidence in the correctness of the general formula.

Lemma 6.3.1 (Translators of LFT d-contractions).
For any $I = [a, b] \in \mathcal{I}$ *and* $L_{r,s,t}|_I \in \mathrm{dLFT}_I$ *with* $s \neq \infty$*, it holds*

$$T_{L_{r,s,t}|_I} = L_{a,b,(s-b)/(s-a)} = \begin{pmatrix} s-a-b & ab \\ -1 & s \end{pmatrix}.$$

(A d-contraction of the form $L_{r,\infty,t}|_I \in \mathrm{dLFT}_I$ *is affine and it, therefore, holds* $T_{L_{r,\infty,t}|_I} = \mathrm{id}_I$*.)*

Proof. Firstly, the function $L_{a,b,(s-b)/(s-a)}$ is a bijection of $[a, b]$ because it is a continuous bijection on $\mathbb{R}^*$, it keeps the endpoints a, b fixed and it maps the point $s \notin [a, b]$ to $\infty \notin [a, b]$.

Secondly, the composition $L_{a,b,(s-b)/(s-a)} \circ L_{r,s,t} \circ L^{-1}_{a,b,(s-b)/(s-a)}$ is affine because the following holds for some expressions in place of each question-mark:

$$\begin{pmatrix} s-a-b & ab \\ -1 & s \end{pmatrix} \cdot \begin{pmatrix} r-ts & (t-1)rs \\ 1-t & tr-s \end{pmatrix} \cdot \begin{pmatrix} s & -ab \\ 1 & s-a-b \end{pmatrix} =$$

$$\begin{pmatrix} s-a-b & ab \\ -1 & s \end{pmatrix} \cdot \begin{pmatrix} ts(r-s) & ? \\ t(r-s) & ? \end{pmatrix} = \begin{pmatrix} ? & ? \\ 0 & ? \end{pmatrix}$$

□

Corollary 6.3.2. *The translator of any non-slow* $f \in \mathrm{dLFT}_I$ *is an LFT.*

Note the fascinating fact is that the translator of a non-slow LFT d-contraction depends only on the base interval and its fixpoint outside the base interval.

Theorem 6.3.3 (Main limitation theorem for LFT-transducers).
Let $\mathcal{T}$ *be a precise finite transducer of type* $(\mathcal{S}_1 \times \cdots \times \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ *with all representations* $\mathcal{S}_i$ *being LFT and open. Any continuously differentiable thread* $h\colon J \to I_0$ *of* $f_{\mathcal{T}}$ *is equal to* $L|_J$ *for some* $\leqslant n$*-ary LFT* L*.*

Proof. By Proposition 6.2.9 all LFT-representations are non-slow on the interior. We can apply Corollary 5.2.15 on $\mathcal{T}$ to obtain that h is in *Glue*$(\mathcal{A}|_J)$. Every element

$$D_{\mathcal{S}_0}(\nu_0)^{-1} \circ T_{D_{\mathcal{S}_0}(u_0)} \circ A \circ \Big(T_{D_{\mathcal{S}_1}(u_1)} \circ D_{\mathcal{S}_1}(\nu_1), \ldots, T_{D_{\mathcal{S}_n}(u_n)} \circ D_{\mathcal{S}_n}(\nu_n)\Big)$$

of $\mathcal{A}$ is a $\left|P^A\right|$-ary LFT by Lemma 6.1.7 because all $D_{\mathcal{S}_i}(u_i)$ and $T_{D_{\mathcal{S}_i}(v_i)}$ are unary LFT's. Since the set of $\leqslant n$-ary LFT's is locally determinable, it holds *Glue*$(\mathcal{A}|_J) = \mathcal{A}|_J$ by Remark 5.2.10, which finally implies that $h = L|_J$ for some $\leqslant n$-ary LFT L. □

Corollary 6.3.4 (Main limitation theorem for extended LFT-transducers).
Let $\mathcal{T}$ be a precise finite transducer of type $(\mathcal{S}_1 \times \cdots \times \mathcal{S}_n) \rightrightarrows \mathcal{S}_0$ with all representations $\mathcal{S}_i$ being extended LFT and open. Any continuously differentiable thread $h\colon J \to I_0$ of $f_{\mathcal{T}}$ is equal to $L|_J$ for some $\leqslant n$-ary LFT L.

Proof. Denote by $\mathcal{S}'_i$ the LFT-representation $(L_i(I_i), \Sigma_i, \mathfrak{d} \mapsto L_i \circ \mathfrak{d} \circ L_i^{-1})$ translated from the extended LFT-representation $\mathcal{S}_i = (I_i, L_i, \Sigma_i, D_i)$ for each $i = 0, \dots, n$. The transducer $\mathcal{T}$ can be considered also of type $(\mathcal{S}'_1 \times \cdots \times \mathcal{S}'_n) \rightrightarrows \mathcal{S}'_0$ because every pair of representations $\mathcal{S}_i$ and $\mathcal{S}'_i$ use the same digit alphabet. Denote the transducer $\mathcal{T}$ with this new type by $\mathcal{T}'$. For any computation $\gamma \in \mathcal{H}_{\mathcal{T}} = \mathcal{H}_{\mathcal{T}'}$, it holds

$$\begin{aligned} D_i(\Pi_i(\gamma)) &= L_i^{-1} \circ D_{\mathcal{S}'_i}(\Pi_i(\gamma)) \circ L_i && \text{for finite } \gamma, \\ [\![\Pi_i(\gamma)]\!]_{\mathcal{S}_i} &= L_i^{-1}\left([\![\Pi_i(\gamma)]\!]_{\mathcal{S}'_i}\right) && \text{for infinite } \gamma. \end{aligned}$$

This implies that $\mathcal{T}'$ is precise and it holds $f_{\mathcal{T}} = L_0^{-1} \circ f_{\mathcal{T}'} \circ (L_1, \dots, L_n)$, i.e. by the preceding theorem, $f_{\mathcal{T}'}$ is an $\leqslant n$-ary LFT. Therefore also $f_{\mathcal{T}}$ is an $\leqslant n$-ary LFT. □

Chapter 7

Conclusion

7.1 Evaluation

In this thesis, we have formalized a particular computational model for real many-valued functions with multiple arguments using a wide class of incremental exact real representations. We have proved a limitation theorem for functions computable with finite memory which covers nearly all of the representations that we have considered. We have also shown some important examples of finitely computable single- and many-valued functions using the signed binary representation. The author hopes that some of the new proof techniques used in this thesis can bring insight to other related areas of research of exact real number computation. It applies especially to the idea of regarding states as functions and the idea of viewing contractions as affine contractions in a different metric.

In Chapter 1, we have mentioned two general questions that our research might help to answer, i.e.

> *Which real number functions can be computed very cheaply?*
>
> *Which exact real number representations offer the most useful range of very cheaply computable functions?*

We can conclude that in the world of affine representations, there are not many interesting 'nicely behaved' functions that can be computed with finite state, i.e. with the cheapest method in the market. For LFT-representations, there is a small hope for a slightly wider class—i.e. n-ary LFT's. However, so far no LFT-representation that can compute significantly more with finite memory than, say, the signed binary representation has been discovered. (This is due to the fact that all of the LFT-representations considered so far were translations of affine representation by one and the same translator (e.g. [Edalat & Potts 1997]) or they were not redundant (e.g. [Raney 1973]).)

7.2 Future work

Although it might seem that we know enough about finite computability to abandon any hope for a surprising discovery, the author still has the hope that one day a significantly better representation might be discovered. Even if there does not exist one, it still might be worthwhile proving it because it would certainly make us understand why it is so and thus deepen our understanding of a part of exact real number computation.

Let us now list several concrete directions in which the results and methods of this thesis could be extended and applied:

1. Inspired by the work of Lisovik and Shkaravskaya [1998], one could strengthen some of the limitation theorems by weakening the precondition of 'continuous differentiability of a thread' to 'differentiability of a thread'.

2. One could almost certainly extend the results to apply also to *push-down automata* in a similar way like Lisovik & Shkaravskaya [1998] and maybe also to the whole of the linear time complexity class.

3. How do the limitation results extend to imprecise finite state computation, i.e. when allowing also incomplete values?

4. Are there any LFT-representations that can finitely compute a class of LFT's which is significantly better than that of Edalat & Potts [1997]?

5. Which functions can IFS-representations with *slow* d-contractions compute with finite memory? How do they compare to those that are computable using the non-slow representations?

6. If we understand the translators T_f from Chapter 5 better, maybe we can find a completely new class of d-contractions that leads to interesting real number representations.

7. Are there any interesting or important higher order functions that can be computed very cheaply? What computational model and representation are best for higher order real number computation? [1]

8. Find a complete axiomatic characterization of first-order multiple-argument many-valued functions expressible by transducers using IFS- or more general representations.

[1] There has been some research in this direction by Weihrauch (see e.g. [1997]), Ko [1991] and Lisovik [1992, 1996].

Bibliography

Abramsky, S. & Jung, A. [1994], Domain theory, *in* S. Abramsky, D. M. Gabbay & T. S. E. Maibaum, eds, 'Handbook of Logic in Computer Science', Vol. 3, Clarendon Press, Oxford, pp. 1–168.

Aubin, J.-P. & Cellina, A. [1984], *Differential inclusions. Set-valued maps and viability theory.*, number 264 *in* 'Grundlehren der Mathematischen Wissenschaften', Springer-Verlag, Berlin.

Aubin, J.-P. & Frankowska, H. [1990], *Set-Valued Analysis*, Birkhaeuser, Boston.

Avizienis, A. [1961], 'Signed-digit number representations for fast parallel arithmetic', *IRE Transactions on electronic computers* **10**, 389–400. Reprinted in E. E. Swartzlander, *Computer Arithmetic*, Vol. 2, IEEE Computer Society Press, Los Alamitos, CA, 1990.

Böhm, H. J. & Cartwright, R. [1990], Exact real numbers: Formulating real numbers as functions, *in* D. A. Turner, ed., 'Research Topics in Functional Programming', Addison-Welsey, Reading, MA, pp. 43–64.

Böhm, H. J., Cartwright, R., Riggle, M. & O'Donnell, M. J. [1986], Exact real arithmetic: A case study in higher order programming, *in* 'Proceedings of the 1986 ACM Conference on LISP and Functional Programming', ACM, New York, NY, pp. 162–173. Held at MIT, Cambridge, MA.

Brattka, V. [1999], Recursive and computable operations over topological structures, Informatik Berichte 255, FernUniversität Hagen, Fachbereich Informatik, Hagen. Dissertation.

Brattka, V. & Hertling, P. [2000], Topological properties of real number representations. Accepted for publication in Theoretical Computer Science.

Diestel, R. [1997], *Graph theory*, Springer-Verlag, New York. Translated from the 1996 German original.

Edalat, A. [1995], 'Dynamical systems, measures and fractals via domain theory', *Information and Computation* **120**(1), 32–48.

Edalat, A. [1996], 'Power domains and iterated function systems', *Information and Computation* **124**(2), 182–197.

Edalat, A. & Potts, P. [1997], 'A new representation of exact real numbers', *Electronical Notes in Theoretical Computer Science* **6**.

Gosper, R. W. [1972], 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`.

Heckmann, R. [1998*a*], The appearance of big integers in exact real arithmetic based on linear fractional transformations, *in* 'Foundations of Software Science and Computation Structures', Vol. 1378 of *Lecture Notes in Computer Science*, Springer Verlag, pp. 172–188.

Heckmann, R. [1998*b*], 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, pp. 45–59. Paris, France, April 27-29, 1998.

Heckmann, R. [2000], Translation of Taylor series into LFT expansions. Submitted.

Hutchinson, J. E. [1981], 'Fractals and self-similarity', *Indiana University Mathematics Journal* **30**(5), 713–747.

Ko, K.-I. [1991], *Complexity Theory of Real Functions*, Birkhäuser, Boston.

Konečný, M. [1998], Real functions incrementally computable by finite automatons, Technical Report CSR-98-7, School of Computer Science, University of Birmingham.

Konečný, M. [2000*a*], Real functions computable by finite automata using affine representations. To be published in Theoretical Computer Science.

Konečný, M. [2000*b*], Real functions computable by finite transducers using affine IFS representations, Technical Report CSR-00-10, School of Computer Science, University of Birmingham.

Kornerup, P. & Matula, D. W. [1995], 'LCF: A lexicographic continued fraction representation of the rationals', *Journal of Universal Computer Science* **1**(7), 480–499.

Kreitz, C. & Weihrauch, K. [1985], 'Theory of representations', *Theoretical Computer Science* **38**(1), 35–53.

Lisovik, L. [1992], 'Macrotransducers on labeled trees and partially continuous operators.', *Cybernetics and Systems Analysis* **28**(1), 11–18.

Lisovik, L. [1996], 'Using macrotransducers to specify partial continuous operators in metric spaces.', *Cybernetics and Systems Analysis* **32**(5), 625–631.

Lisovik, L. P. [1987], 'Algorithmic questions for real functions', *Kibernetika* **23**(1), 12–17. In Russian.

Lisovik, L. P. [1990], 'Classes of functions defined by transducers', *Kibernetika* **26**(6), 1–7. In Russian.

Lisovik, L. P., Koval, D. A. & Martinez, S. V. [1999], 'R*-transducers and fractal curves', *Kibernetika i Sistemnyi Analiz (formerly Kibernetika)* (3), 95–105. In Russian.

Lisovik, L. P. & Shkaravskaya, O. Y. [1995], 'Functions defined by push-down transducers', *Dopovidi Natsionalnoji Akademiji Nauk Ukrainy* (9), 57–59. In Russian.

Lisovik, L. P. & Shkaravskaya, O. Y. [1998], 'About real functions defined by transducers', *Kibernetika i Sistemnyi Analiz (formerly Kibernetika)* (1), 82–93. In Russian.

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

Ménissier-Morain, V. [1996], 'Arbitrary precision real arithmetic: design and algorithms', Submitted to the Journal of Symbolic Computation.

Ore, O. [1962], *Theory of Graphs*, American Mathematical Society, Providence, RI.

Peyton Jones, S. L. [1984], Arbitrary precision arithmetic using continued fractions, INDRA Note 1530, Department of Computer Science, University College, London, UK.

Potts, P. & Edalat, A. [1997], Exact real computer arithmetic, Technical Report DOC 97/9, Department of Computer Science, Imperial College.

Potts, P. J. [1998], Computable Real Arithmetic Using Linear Fractional Transformations, PhD thesis, Imperial College, Department of Computing.

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

Rudin, W. [1953], *Principles of Mathematical Analysis*, McGraw-Hill, New York, NY.

Rudin, W. [1986], *Real and Complex Analysis*, 3 edn, McGraw-Hill, New York, NY.

Schönhage, A. [1982], 'The fundamental theorem of algebra in terms of computational complexity', preprint. Mathematishes Institut der Universität Tübingen.

Shkaravskaya, O. Y. [1998], 'On affine mapping defined by finite transducers', *Kibernetika i Sistemnyi Analiz (formerly Kibernetika)* (5), 178–181. In Russian.

Smyth [1992], Topology, *in* S. Abramsky, D. M. Gabbay & T. S. E. Maibaum, eds, 'Handbook of Logic in Computer Science', Vol. 1, Clarendon Press, Oxford, pp. 641–761.

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

Turing, A. M. [1935], 'On computable numbers with an application to the Entscheidungsproblem', *Proceedings of London Mathematical Society. II. Series.* **42**, 230–265.

Turing, A. M. [1937], 'On computable numbers with an application to the Entscheidungsproblem. a correction', *Proceedings of London Mathematical Society. II. Series.* **43**, 544–546.

Vietoris, L. [1922], 'Beraiche zweiter ordnung', *Monatshefte Mathematik und Physik* **32**, 258–280.

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

Weihrauch, K. [1987], *Computability*, number 9 *in* 'EATCS Monographs on Theoretical Computer Science', Springer, Berlin.

Weihrauch, K. [1995], A simple introduction to computable analysis, Informatik Berichte 171, FernUniversität Hagen, Hagen. 2nd edition.

Weihrauch, K. [1997], A foundation for computable analysis, *in* D. S. Bridges, C. S. Calude, J. Gibbons, S. Reeves & I. H. Witten, eds, 'Combinatorics, Complexity, and Logic', DMTCS'96: Discrete Mathematics and Theoretical Computer Science, Springer, Singapore, 9-13 December 1996, Auckland, New Zealand, pp. 66–89.

Wiedmer, E. [1980], 'Computing with infinite objects', *Theoretical Computer Science* **10**(2), 133–155.

Symbols and Notations

Sequences (words)

$\mathbb{N}$	the set of natural numbers
n, m, i, j, k, ℓ	natural numbers
Σ	finite set of *symbols*
ϵ	the empty sequence
Σ^n	the set of sequences of length n over Σ
Σ^*	the set of finite sequences over Σ
u, v	finite sequences
$\lvert u \rvert$	the length of a sequence u
Σ^ω	the set of infinite sequences over Σ
Σ^∞	$= \Sigma^* \cup \Sigma^\omega$
α, β, γ	infinite or finite sequences
$\lvert \alpha \rvert$	$= \infty$ for infinite α
$\alpha[j]$	the j-th symbol of α
$\alpha[j, k]$	the sequence $\alpha[j]\alpha[j+1]\cdots\alpha[k]$
$\alpha[j, \infty]$	the infinite sequence $\alpha[j]\alpha[j+1]\cdots$
$\alpha[\infty, \infty]$	$= \epsilon$
$\alpha \sqsubseteq \beta$	iff $\alpha = \beta[1, k]$ or $\alpha = \beta$
$\beta \setminus \alpha$	$\beta[k+1, \lvert\beta\rvert]$ if $\alpha = \beta[1, k]$, ϵ if $\alpha = \beta$

Sets

X, Y, Z	arbitrary sets
x, y, z	arbitrary elements
$U, J \ldots$	subsets of a space
$\mathcal{U}, \mathcal{K}$	collections of subsets of a space
$\bigcup \mathcal{U}$	$= \bigcup_{U \in \mathcal{U}} U$
$\overline{U}$	closure of U
U^o	interior of U

Functions

f, g, h		functions
$f\colon X \to Y$		(f is a) function (from X to Y)
$f\colon X \rightharpoonup Y$		partial function
$f\colon X \rightrightarrows Y$	*[p13]*	partial many-valued function
$f\colon X \rightrightarrows Y$	*[p13]*	total many-valued function
$\mathbf{Gr}(f)$	*[p13]*	the graph of f
$f \subseteq g$		$\mathrm{Gr}(f) \subseteq \mathrm{Gr}(g)$
$\mathbf{id}_X$		the identity on X
f^{-1}	*[p13]*	the inversion of f
$g \circ f$	*[p13]*	function composition: g after f
f^n		iteration $f \circ \cdots \circ f$ (n times)
$f\vert_U$		restriction of f to domain U
$f\vert^U$		corestriction of f to codomain U
$\mathcal{F}$	*[p84]*	set of functions of the same type
Glue$(\mathcal{F})$	*[p85]*	the gluing closure of $\mathcal{F}$

Real Analysis

$\mathbb{R}$		the space of real numbers
$r, s, \varepsilon, \delta, a, b, c, d$		real numbers
$\mathcal{I}$	*[p13]*	the set of all closed non-singleton intervals in $\mathbb{R}$

$\mathbf{I} = [\mathbf{a}, \mathbf{b}]$ *[p100]* closed non-singleton interval

(a, b) open interval

$(a, b], [a, b)$ intervals open on one side and closed on the other side

$\mathbf{neg_I}$ *[p90]* the decreasing affine bijection $x \mapsto (a + b - x) : I \to I$

$\mathcal{I}(\mathbf{I})$ *[p26]* the set of all closed subintervals within I

$\mathbb{R}^*$ *[p17]* $= \mathbb{R} \cup \{\infty\}$, the set of extended real numbers

$\mathcal{I}^*$ *[p100]* all closed non-singleton extended real intervals

$\vec{x}, \ldots$ a tuple of elements

$\vec{f}, \ldots$ a tuple of functions

$\mathbf{V}$ *[p20]* $= I_1 \times \cdots \times I_n$

$\mathbf{P}$ *[p20]* subset of $\{1, \ldots, n\}$

$\overline{\mathbf{P}}$ *[p20]* the set $\{1, \ldots, n\} \setminus P$

$\vec{x}_{\mathbf{P}}$ *[p20]* the projection of $\vec{x}$ to coordinates P

$f_{\mathbf{P}, x_{\overline{\mathbf{P}}}}$ *[p20]* the instantiation of f at coordinates P by $x_{\overline{P}}$

$|\vec{x} - \vec{y}|$ *[p20]* Euclidean distance of $\vec{x}$ and $\vec{y}$

$\mathbf{B}_\varepsilon(\vec{x})$ *[p70]* $= [x_1 - \varepsilon, x_1 + \varepsilon] \times \cdots \cdots \times [x_n - \varepsilon, x_n + \varepsilon]$

IFS-representations

$\mathbf{Fix_f}$ *[p26]* the fixpoint of a d-contraction f

$\mathbf{Strg_f}$ *[p26]* $= f'(\mathrm{Fix}_f)$, the strength of a d-contraction f

$\overline{\mathbf{f}}$ *[p26]* the extension of a d-contraction f to intervals

$\mathbf{dContr_I}$ *[p26]* all d-contractions on I

$\mathbf{dAff_I}$ *[p26]* all affine d-contractions on I

$\mathbf{dLFT_I}$ *[p104]* all LFT d-contractions on I

$\mathcal{S} = (\mathbf{I}, \mathbf{\Sigma}, \mathbf{D})$ *[p30]* IFS-representation

$\mathfrak{d}$ *[p30]* a digit from Σ

$\mathbf{D}(\mathfrak{d})$ *[p29]* the d-contraction of $\mathfrak{d}$

$\mathbf{D}(\mathfrak{d}_1 \cdots \mathfrak{d}_n)$ *[p33]* $= D(\mathfrak{d}_1) \circ \cdots \circ D(\mathfrak{d}_n)$

$[\![\mathbf{u}]\!]$ *[p30]* $= \big(D(u)\big)(I)$

$[\![\boldsymbol{\alpha}]\!]$ *[p30]* $= \bigcap_{k \leqslant |\alpha|} [\![\alpha[1, k]]\!]$

$\mathbf{Adic}_{\mathcal{S}}$ *[p34]* all $\mathcal{S}$-adic numbers

$\mathbf{LAdic}_{\mathcal{S}}$ *[p34]* all left $\mathcal{S}$-adic numbers

$\mathbf{RAdic}_{\mathcal{S}}$ *[p34]* all right $\mathcal{S}$-adic numbers

$\mathbf{Adic}_{\mathcal{S}}^{\leqslant \ell}$ *[p34]* all $\mathcal{S}$-adic numbers of order $\leqslant \ell$

$\mathbf{T_f}$ *[p76]* the stretching translator of d-contraction f

$\mathbf{A_f}$ *[p76]* the stretched translation of d-contraction f

Transducers

$\mathcal{T} = (\mathbf{Q}, \zeta, \mathbf{q_I})$ *[p36]* transducer

$\mathbf{Q}$ *[p36]* set of states

ζ *[p36]* transition function

$\mathbf{q_I}$ *[p36]* initial state

$\mathcal{H}_{\mathcal{T}}^{\mathbf{fin}}$ *[p37]* all finite computations of $\mathcal{T}$

$\mathcal{H}_{\mathcal{T}}$ *[p37]* all finite and infinite computations of $\mathcal{T}$

$\boldsymbol{\gamma}$ *[p37]* computation

$\boldsymbol{\gamma}|_{\mathbf{P}}$ *[p37]* the projection of γ to channels P

$\boldsymbol{\Pi}_{\mathbf{In}}(\boldsymbol{\gamma})$ *[p37]* total input of computation γ

$\boldsymbol{\Pi}_{\mathbf{Out}}(\boldsymbol{\gamma}) = \boldsymbol{\Pi}_0(\boldsymbol{\gamma})$ *[p37]* total output of computation γ

$\mathbf{q}_{\boldsymbol{\gamma}}$ *[p37]* the final state of a finite γ

$\boldsymbol{\sigma}_{\mathcal{T}}$ *[p37]* the stream function computed by $\mathcal{T}$

$\overline{\mathbf{f}}_{\mathcal{T}}$ *[p40]* the interval valued function computed by IFS-transducer $\mathcal{T}$

$\mathbf{f}_{\mathcal{T}}$ *[p41]* the real function computed by IFS-transducer $\mathcal{T}$

$\mathcal{G}_{\mathcal{T}}$ *[p38]* the underlying graph of $\mathcal{T}$

Index