



Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

SEN

Software Engineering



Software ENgineering

Elements of stream calculus
(an extensive exercise in coinduction)

J.J.M.M. Rutten

REPORT SEN-R0120 JULY 2001

CWI is the National Research Institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organization for Scientific Research (NWO).

CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

Copyright © 2001, Stichting Centrum voor Wiskunde en Informatica

P.O. Box 94079, 1090 GB Amsterdam (NL)

Kruislaan 413, 1098 SJ Amsterdam (NL)

Telephone +31 20 592 9333

Telefax +31 20 592 4199

ISSN 1386-369X

Elements of Stream Calculus (an extensive exercise in coinduction)

J.J.M.M. Rutten

CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

Email: `janr@cw.nl`, URL: `www.cwi.nl/~janr`

ABSTRACT

Based on the presence of a final coalgebra structure on the set of streams (infinite sequences of real numbers), a coinductive calculus of streams is developed. The main ingredient is the notion of stream derivative, with which both coinductive proofs and definitions can be formulated. In close analogy to classical analysis, the latter are presented as behavioural differential equations. A number of applications of the calculus are presented, including difference equations, analytical differential equations, continued fractions, and some problems from discrete mathematics and combinatorics.

2000 Mathematics Subject Classification: 68Q10, 68Q55, 68Q85

1998 ACM Computing Classification System: F.1, F.3

Keywords & Phrases: Coinduction, coalgebra, finality, stream, derivative, differential equation, difference equation, Euler formula, counting, formal power series, weighted automaton

Contents

1	Introduction	3
2	Streams and coinduction	4
3	Behavioural differential equations	6
4	Basic stream calculus	8
5	Solving behavioural differential equations	17
6	Application: solving difference equations	19
7	Solving quadratic equations in stream calculus	21
8	Shuffle product and shuffle inverse	24
9	Application: a divergent recurrence	27
10	Comparing convolution product and shuffle product	28
11	Application: a generalised Euler formula	31
12	Application: solving analytical differential equations	32
13	Weighted stream automata	37
14	Multivariate streams	43
15	Multivariate stream calculus	45
16	Weighted automata for multivariate streams	46
17	Application: coinductive counting	47
18	Discussion and related work	50
19	Appendix	52

1 Introduction

Treating infinite sequences of real numbers, called *streams* here, as single entities, a *calculus* of streams is developed in two ways:

1. as in analysis, involving stream differentiation and integration;
2. as in algebra, where one calculates with operators, and establishes identities.

Starting point is the fact that the set $\mathbb{R}^\omega$ of all streams carries a *final coalgebra structure*, consisting of the following pair of operations:

$$\begin{array}{ccc} \mathbb{R}^\omega & & \sigma \\ \langle O, T \rangle \downarrow & & \downarrow \\ \mathbb{R} \times \mathbb{R}^\omega & & \langle \sigma(0), \sigma' \rangle \end{array}$$

They assign to a stream $\sigma = (s_0, s_1, s_2, \dots)$ its *initial value* $\sigma(0) \in \mathbb{R}$ and its *stream derivative* $\sigma' \in \mathbb{R}^\omega$, which are defined by

$$\begin{aligned} \sigma(0) &= s_0 \\ \sigma' &= (s_1, s_2, s_3, \dots) \end{aligned}$$

(In computer science, these operators are usually referred to as *head* and *tail*.) Being a final coalgebra, the set $\mathbb{R}^\omega$ satisfies a *coinduction proof principle* and a *coinduction definition principle*, both formulated in terms of the notions of initial value and stream derivative. We shall think of and deal with the operation of stream derivation in close analogy to classical analysis. Notably, coinductive definitions take the shape of what we have called *behavioural differential equations*, defining streams and stream operators by means of equations that involve initial values and derivatives of streams (see Section 3 for some first elementary examples). By means of such behavioural differential equations, many stream operators are introduced, some familiar (such as sum, convolution product, shuffle product) and some less familiar (for instance, shuffle inverse, stream exponentiation, square root). They are summarised in the following table:

operator	name
$\sigma(0)$	initial value
σ'	stream derivative
r	constant ($r \in \mathbb{R}$)
X	formal variable
$\sigma + \tau$	sum
$\sigma \times \tau$	convolution product
$\sum_{n=0}^{\infty} \sigma_n$	infinite sum
σ^{-1}	(multiplicative) inverse
$\sigma \circ \tau$	stream composition
$\sqrt{\sigma}$	square root
$\sigma \otimes \tau$	shuffle product
σ^{-1}	shuffle inverse
$\sigma^{1/2}$	shuffle square root
$\exp(\sigma)$	stream exponentiation
$\frac{d\sigma}{dX}$	analytical stream derivative
$\Lambda_c(\sigma)$	Laplace-Carson transform

A large number of basic and somewhat more advanced facts are proved. Among the most interesting ones are the identities for stream exponentiation and for *shuffle elimination* (Sections 8 and 10). All our reasoning, that is, almost all definitions and proofs, will be coinductive.

In addition to the development of the stream calculus itself, some applications of the calculus are presented, relating to a number of different disciplines in mathematics:

- discrete mathematics: solving difference equations (Sections 6 and 9) and a generalised Euler formula (Section 11);
- analysis: solving differential equations (Section 12), and continued fractions (Section 17);
- combinatorics: coinductive counting (Section 17).

In the ongoing work on coalgebra, the present paper can be seen as a study in *concrete* coalgebra, as opposed to the more general and abstract studies of *universal coalgebra*. Another way of putting this, as expressed by the subtitle of this paper, is to view stream calculus as a study of ‘coinduction at work’. We have tried to make the paper as self-contained as possible. In particular, everything the reader needs to know about coalgebra and coinduction, will be explicitly introduced.

In conclusion of this introduction, we briefly summarise related work, which will be discussed more extensively in Section 18.

- The present paper builds on [Rut00a], by repeating some of its basics, extending its results on streams, and adding the applications mentioned above.
- An important source of inspiration has been [PE98], both for the guiding role of analysis in the development of the present calculus, and for the application to analytical differential equations.
- Generating functions are a classical tool in mathematics for reasoning about streams; [Wil94] and [GKP94] are two inspiring references. We believe coinductive stream calculus to be an alternative to the calculus of generating functions, being both more formal (essentially because of a rigid use of coinduction) and more expressive (because of the presence of a number of non-standard stream operators, such as shuffle inverse).
- Formal power series (see [BR88]), which are often taken as the more formal version of generating functions, occur in the present paper towards the end (Section 14) under the name of multivariate streams. Again, we see some advantages in the use of some of the non-standard stream operators presented here, as well as in the use of the coinduction definition and proof principles.
- Streams are a canonical example in the paradigm of lazy functional programming. Because of this close connection, behavioural differential equations can be interpreted as effective, directly implementable recipes for the step-wise generation of streams. Both [McI99, McI01] and [Kar97, Kar00] have served as very enjoyable sources of examples. One thing that coinductive stream calculus seems to add to this type of lazy functional programming, is coinduction as a systematic way of reasoning about streams.

2 Streams and coinduction

The set $\mathbb{R}^\omega$ of all streams is introduced. Using some elementary notions from universal coalgebra, $\mathbb{R}^\omega$ is shown to satisfy a proof principle called coinduction. Moreover, $\mathbb{R}^\omega$ is shown to have the universal property of being a final stream automaton (coalgebra). Finality will be the basis for coinductive definition schemata, in Section 3.

The set of all streams is formally defined by

$$\mathbb{R}^\omega = \{\sigma \mid \sigma : \{0, 1, 2, \dots\} \rightarrow \mathbb{R}\}$$

We shall call $\sigma(0)$ the *initial value* of σ . The *derivative* of a stream σ is defined, for all $n \geq 0$, by

$$\sigma'(n) = \sigma(n + 1)$$

Although streams, which are infinite sequences of real numbers, will be viewed and handled as single mathematical entities, it will at various moments be convenient to refer to the individual elements of which they are made. For this, we shall use the following notation:

$$\begin{aligned}\sigma &= (\sigma(0), \sigma(1), \sigma(2), \dots) \\ &= (s_0, s_1, s_2, \dots)\end{aligned}$$

(Similarly, we shall write $\tau = (t_0, t_1, t_2, \dots)$ and the like.) With this notation, the derivative of σ is given by

$$\sigma' = (s_1, s_2, s_3, \dots)$$

For any $n \geq 0$, the real number $\sigma(n) = s_n$ is called the n -th *element* of σ . It can also be expressed in terms of higher-order stream derivatives, defined, for all $k \geq 0$, by

$$\sigma^{(0)} = \sigma, \quad \sigma^{(k+1)} = (\sigma^{(k)})'$$

Now the n -th element of σ is also given by

$$\sigma(n) = \sigma^{(n)}(0) \tag{1}$$

In order to conclude that two streams σ and τ are equal, it is both necessary and sufficient to prove

$$\forall n \geq 0, \sigma(n) = \tau(n) \tag{2}$$

What general methods for establishing the validity of (2) are there? A straightforward *induction* on the natural number n (prove $\sigma(0) = \tau(0)$ and show that $\sigma(n) = \tau(n)$ implies $\sigma(n+1) = \tau(n+1)$) may seem the obvious answer, but very often it is not. There will be numerous occasions where we will have no workable description or formula for $\sigma(n)$ and $\tau(n)$, and where, consequently, induction simply cannot be applied.

Instead, we shall take a coalgebraic perspective on $\mathbb{R}^\omega$, and use almost exclusively the proof principle of *coinduction*, which is formulated in terms of the following notion from the world of universal coalgebra. A *bisimulation* on $\mathbb{R}^\omega$ is a relation $R \subseteq \mathbb{R}^\omega \times \mathbb{R}^\omega$ such that, for all σ and τ in $\mathbb{R}^\omega$,

$$\text{if } \sigma R \tau \text{ then } \begin{cases} \sigma(0) = \tau(0) & \text{and} \\ \sigma' R \tau' \end{cases}$$

(Here $\sigma R \tau$ denotes $\langle \sigma, \tau \rangle \in R$; both notations will be used.) One easily checks that unions and (relational) compositions of bisimulations are bisimulations again. We write $\sigma \sim \tau$ whenever there exists a bisimulation R with $\sigma R \tau$. This relation $\sim$, called the *bisimilarity* relation, is the union of all bisimulations and, therewith, the greatest bisimulation.

Theorem 2.1 [Coinduction] *If two streams σ and τ are bisimilar: $\sigma \sim \tau$, then it follows that $\sigma(n) = \tau(n)$, for all $n \geq 0$, and, consequently, $\sigma = \tau$. That is, for all $\sigma, \tau \in \mathbb{R}^\omega$,*

$$\sigma \sim \tau \Rightarrow \sigma = \tau$$

(Note that the converse trivially holds, since $\{\langle \sigma, \sigma \rangle \mid \sigma \in \mathbb{R}^\omega\}$ is a bisimulation relation on $\mathbb{R}^\omega$.) Thus in order to prove the equality of two streams σ and τ , it is sufficient to establish the existence of a bisimulation relation $R \subseteq \mathbb{R}^\omega \times \mathbb{R}^\omega$ with $\langle \sigma, \tau \rangle \in R$.

Proof: Consider two streams σ and τ and let $R \subseteq \mathbb{R}^\omega \times \mathbb{R}^\omega$ be a bisimulation on $\mathbb{R}^\omega$ containing the pair $\langle \sigma, \tau \rangle$. It follows by induction on n that $\langle \sigma^{(n)}, \tau^{(n)} \rangle \in R$, for all $n \geq 0$, because R is a bisimulation. This implies, again because R is a bisimulation, that $\sigma^{(n)}(0) = \tau^{(n)}(0)$, for all $n \geq 0$. By identity (1), $\sigma(n) = \tau(n)$, for all $n \geq 0$. Now $\sigma = \tau$ follows. $\square$

We shall see many examples of proofs by coinduction: one of the main reasons why we have become interested in $\mathbb{R}^\omega$ at all, is that it offers a perfect playground for demonstrating the use and usefulness of coinduction.

The word coinduction is also used as a term for certain definitions. This will be the subject of Section 3, where coinductive definitional schemata are introduced in terms of so-called behavioural differential equations. Such equations can be shown to have unique solutions on the basis of a universal property of $\mathbb{R}^\omega$, which we introduce next, again using a little bit of elementary universal coalgebra.

A stream coalgebra or automaton is a pair $Q = (Q, \langle o, t \rangle)$ consisting of a set Q of states, together with an output function $o : Q \rightarrow \mathbb{R}$, and a transition function $t : Q \rightarrow Q$. A homomorphism between stream automata $(Q, \langle o, t \rangle)$ and $(Q', \langle o', t' \rangle)$ is a function $f : Q \rightarrow Q'$ such that, for all q in Q , $o(q) = o'(f(q))$ and $f(t(q)) = t'(f(q))$ or, in other words, such that the following diagram commutes:

$$\begin{array}{ccc} Q & \xrightarrow{f} & Q' \\ \langle o, t \rangle \downarrow & & \downarrow \langle o', t' \rangle \\ \mathbb{R} \times Q & \xrightarrow{1_{\mathbb{R}} \times f} & \mathbb{R} \times Q' \end{array}$$

(The function $1_{\mathbb{R}} \times f$ maps a pair $\langle r, q \rangle \in \mathbb{R} \times Q$ to $\langle r, f(q) \rangle \in \mathbb{R} \times Q'$.)

The set $\mathbb{R}^\omega$ of all streams can itself be turned into a stream automaton. Defining $O : \mathbb{R}^\omega \rightarrow \mathbb{R}$ by $O(\sigma) = \sigma(0)$ and $T : \mathbb{R}^\omega \rightarrow \mathbb{R}^\omega$ by $T(\sigma) = \sigma'$, we obtain a stream automaton $(\mathbb{R}^\omega, \langle O, T \rangle)$. It has the following universal property.

Theorem 2.2 *The automaton $(\mathbb{R}^\omega, \langle O, T \rangle)$ is final among the family of all stream automata. That is, for any automaton $(Q, \langle o, t \rangle)$ there exists a unique homomorphism $l : Q \rightarrow \mathbb{R}^\omega$:*

$$\begin{array}{ccc} Q & \xrightarrow{\exists! l} & \mathbb{R}^\omega \\ \langle o, t \rangle \downarrow & & \downarrow \langle O, T \rangle \\ \mathbb{R} \times Q & \xrightarrow{1_{\mathbb{R}} \times l} & \mathbb{R} \times \mathbb{R}^\omega \end{array}$$

Proof: Let $(Q, \langle o, t \rangle)$ be an automaton and let the function $l : Q \rightarrow \mathbb{R}^\omega$ assign to a state q in Q the stream $(o(q), o(t(q)), o(t(t(q))), \dots)$. It is straightforward to show that l is a homomorphism from $(Q, \langle o, t \rangle)$ to $(\mathbb{R}^\omega, \langle O, T \rangle)$. For uniqueness, suppose f and g are homomorphisms from Q to $\mathbb{R}^\omega$. The equality of f and g follows by coinduction from the fact that $R = \{\langle f(q), g(q) \rangle \mid q \in Q\}$ is a bisimulation on $\mathbb{R}^\omega$, which is proved next. Consider $\langle f(q), g(q) \rangle \in R$. Because f and g are homomorphisms, $O(f(q)) = o(q) = O(g(q))$. Furthermore, $T(f(q)) = f(t(q))$ and $T(g(q)) = g(t(q))$. Because $\langle f(t(q)), g(t(q)) \rangle \in R$, this shows that R is a bisimulation. Thus $f(q) \sim g(q)$, for any q in Q . Now $f = g$ follows by the coinduction proof principle Theorem 2.1. $\square$

3 Behavioural differential equations

In stream calculus, streams and operators on streams will be defined by means of (systems of) *behavioural differential equations*, specifying their derivatives and initial values. Much of the theory of behavioural differential equations has been extensively dealt with in [Rut00a]. Here we shall summarise the main ideas by treating two typical examples, and refer the reader to [Rut00a] for more details. Moreover, as an appendix, Section 19 contains a theorem stating the unique existence of solutions for a rather general family of behavioural differential equations, comprising about all the equations actually encountered in the present paper.

As a first and very elementary example, we prove that there exists a unique stream σ satisfying the following behavioural differential equation:

behavioural differential equation	initial value
$\sigma' = \sigma$	$\sigma(0) = 1$

This example is so simple that the solution is immediate: take $\sigma = (1, 1, 1, \dots)$ and note that, indeed, $\sigma' = \sigma$ and that $\sigma(0) = 1$. Showing that this is the only solution should not be too difficult, either. But just as an exercise, let us try to base our argument solely on coinduction and the finality of $\mathbb{R}^\omega$. We shall benefit from this experience in the next example, which is no longer trivial. In order to find a solution, we define an automaton $(S, \langle o, t \rangle)$, which contains one state: $S = \{s\}$. The automaton structure $\langle o, t \rangle$ is defined next in such a manner that this state behaves as the solution σ should behave according to the differential equation. Thus we define $t : S \rightarrow S$ by $t(s) = s$ and $o : S \rightarrow \mathbb{R}$ by $o(s) = 1$. By the finality of the automaton $(\mathbb{R}^\omega, \langle O, T \rangle)$ (Theorem 2.2), there exists a unique homomorphism $l : S \rightarrow \mathbb{R}^\omega$. We can now define $\sigma = l(s)$. Because l is a homomorphism, $\sigma' = T(\sigma) = T(l(s)) = l(t(s)) = l(s) = \sigma$, and $\sigma(0) = O(\sigma) = O(l(s)) = o(s) = 1$, indeed. Thus we have found a solution of our behavioural differential equation. If ρ is a stream satisfying $\rho' = \rho$ and $\rho(0) = 1$, then $\sigma = \rho$ follows, by the coinduction proof principle Theorem 2.1, from the fact that $\{\langle \sigma, \rho \rangle\}$ is a bisimulation relation of $\mathbb{R}^\omega$. Which shows that σ is the only solution of the differential equation.

So this was easy and proving the unique existence of a solution for any of the equations we shall encounter will, in essence, be just as easy. The only difference will be that the design of the automaton $(S, \langle o, t \rangle)$ generally requires a bit more work (for one thing, the state space S usually is infinite). All of this is very clearly illustrated by our second example, which defines the binary operators of sum and shuffle product. These operators will be motivated and discussed in all detail later. Without having any idea what these operators are about, here we just prove, mechanically as it were, that they are well-defined, again by exploiting coinduction and the finality of $\mathbb{R}^\omega$.

We shall prove that for any two streams σ and τ , there exist unique streams $\sigma + \tau$ and $\sigma \otimes \tau$, satisfying the following system of behavioural differential equations:

behavioural differential equation	initial value
$(\sigma + \tau)' = \sigma' + \tau'$	$(\sigma + \tau)(0) = \sigma(0) + \tau(0)$
$(\sigma \otimes \tau)' = (\sigma' \otimes \tau) + (\sigma \otimes \tau')$	$(\sigma \otimes \tau)(0) = \sigma(0) \times \tau(0)$

Note that it is not so easy now to see in one glance how the solutions should look like. Still, the equations can be rather straightforwardly interpreted as recipes for the construction of the respective elements of the solutions. For instance,

$$\begin{aligned}
(\sigma \otimes \tau)(0) &= \sigma(0) \times \tau(0) \\
(\sigma \otimes \tau)(1) &= (\sigma \otimes \tau)'(0) \\
&= (\sigma' \otimes \tau + \sigma \otimes \tau')(0) \\
&= \sigma'(0) \times \tau(0) + \sigma(0) \times \tau'(0) \\
&= \sigma(1) \times \tau(0) + \sigma(0) \times \tau(1) \\
(\sigma \otimes \tau)(2) &= (\sigma \otimes \tau)''(0) \\
&= (\sigma' \otimes \tau + \sigma \otimes \tau')'(0) \\
&= \dots \\
&= \sigma(2) \times \tau(0) + 2 \times \sigma(1) \times \tau(1) + \sigma(0) \times \tau(2)
\end{aligned}$$

and so on. This illustrates the algorithmic aspect of the behavioural differential equations. A formal proof of the unique existence of their solutions, however, can be much better (and more generally) based on the finality of $\mathbb{R}^\omega$, as follows. We construct what could be called a syntactic stream automaton, whose states are given by expressions including all the possible shapes that occur on the right side of the behavioural differential equations above. The solutions are then given by the unique homomorphism into $\mathbb{R}^\omega$. More precisely, let the set $\mathcal{E}$ of expressions be given by the following syntax:

$$E ::= \underline{\sigma} \mid E + F \mid E \otimes F$$

The set $\mathcal{E}$ contains for every stream σ in $\mathbb{R}^\omega$ a constant symbol $\underline{\sigma}$. The operators that we are in the process of defining, are represented in $\mathcal{E}$ by a syntactic counterpart, denoted by the same symbols $+$ and $\otimes$ again. The set $\mathcal{E}$ is next supplied with an automaton structure $(\mathcal{E}, \langle o, t \rangle)$ by defining functions $o : \mathcal{E} \rightarrow \mathbb{R}$ and $t : \mathcal{E} \rightarrow \mathcal{E}$. For the definition of o and t on the constant symbols $\underline{\sigma}$, the automaton structure $(\mathbb{R}^\omega, \langle O, T \rangle)$ on $\mathbb{R}^\omega$ is used:

$$t(\underline{\sigma}) = \underline{T(\sigma)} = \underline{\sigma'}, \quad o(\underline{\sigma}) = O(\sigma) = \sigma(0)$$

Thus the constant $\underline{\sigma}$ behaves in the automaton $\mathcal{E}$ precisely as the stream σ behaves in the automaton $\mathbb{R}^\omega$. (This includes $\mathbb{R}^\omega$ as a subautomaton in $\mathcal{E}$.) For composite expressions, the definitions of o and t literally follow the definition of the corresponding behavioural differential equations. Writing $E(0)$ for $o(E)$ and E' for $t(E)$, for any expression E in $\mathcal{E}$, we put

definition of t	definition of o
$(E + F)' = E' + F'$	$(E + F)(0) = E(0) + F(0)$
$(E \otimes F)' = (E' \otimes F) + (E \otimes F')$	$(E \otimes F)(0) = E(0) \times F(0)$

The above defines the functions o and t , by induction on the structure of the expressions. Since $\mathcal{E}$ now has been turned into an automaton $(\mathcal{E}, \langle o, t \rangle)$, and because $\mathbb{R}^\omega$ is a final automaton, there exists, by Theorem 2.2, a unique homomorphism $l : \mathcal{E} \rightarrow \mathbb{R}^\omega$, which assigns to each expression E the stream $l(E)$ it represents. It can be used to define the operators on $\mathbb{R}^\omega$ that we are looking for, as follows:

$$\sigma + \tau = l(\underline{\sigma} + \underline{\tau}), \quad \sigma \otimes \tau = l(\underline{\sigma} \otimes \underline{\tau})$$

Using the coinduction proof principle (Theorem 2.1), these operators can now be shown to be the unique solutions of the behavioural differential equations above.

The above two operators are the only ones in the present paper for which we provide a proof of well-definedness. From now on, we shall without further ado introduce all kinds of streams and operators by means of behavioural differential equations. All of them can be justified in the same manner as above, by coinduction and finality of $\mathbb{R}^\omega$. A rather general theorem is presented in appendix Section 19, stating the well-definedness of a large class of (systems of) behavioural differential equations.

Taking the existence of solutions of behavioural differential equations from now on for granted, much energy will still be spent on the computation of closed formulae for such solutions, expressing them in terms of a basic set of stream constants and stream operators. We shall see, for instance, that the solution $\sigma = (1, 1, 1, \dots)$ of the first example above, can be expressed by means of the following formula:

$$\sigma = \frac{1}{1 - X}$$

which will be derived from the defining behavioural differential equation in an algebraic fashion. But now we are getting ahead of our story.

4 Basic stream calculus

A number of basic stream operators is introduced, by means of behavioural differential equations, and some elementary properties and examples are discussed. The presentation of the various coinductive proofs in this section will at moments seem tediously detailed, but along the way, we shall establish some notions and results that will be very helpful in future, less elementary situations. An example is the introduction of the notion of bisimulation-up-to.

First of all, we want to be able to view any real number as a constant stream. For $r \in \mathbb{R}$, let $[r]$ be the unique stream satisfying the following behavioural differential equation:

behavioural differential equation	initial value	name
$[r]' = [0]$	$[r](0) = r$	constant

Hadn't we been obsessed with the use of our behavioural differential equations, we would have given the following definition, which is clearly equivalent:

$$[r] = (r, 0, 0, 0, \dots)$$

Having once formally introduced the inclusion of the reals into the set of streams by means of this operator $[\] : \mathbb{R} \rightarrow \mathbb{R}^\omega$, we immediately observe that there will in stream calculus hardly be any chance of confusing the stream $[r]$ with the real number r it represents. Therefore we shall usually simply write r for $[r]$.

The following equation defines one more constant. Let X be the unique stream satisfying

behavioural differential equation	initial value	name
$X' = [1]$	$X(0) = 0$	(formal) variable

The constant X plays a crucial role in stream calculus, and may be thought of as a formal variable. Again, there is a more explicit equivalent definition:

$$X = (0, 1, 0, 0, 0, \dots)$$

Next we repeat the definition of the operation of sum that we have already seen as an example in Section 3. The sum $\sigma + \tau$ of two streams σ and τ is defined as the unique stream satisfying:

behavioural differential equation	initial value	name
$(\sigma + \tau)' = \sigma' + \tau'$	$(\sigma + \tau)(0) = \sigma(0) + \tau(0)$	sum

(Note that we are using the same symbol $+$ both for the sum of streams and the sum of real numbers.) Alternatively and equivalently, the sum of $\sigma = (s_0, s_1, s_2, \dots)$ and $\tau = (t_0, t_1, t_2, \dots)$ is given by

$$\sigma + \tau = (s_0 + t_0, s_1 + t_1, s_2 + t_2, \dots)$$

Sum satisfies the following familiar identities: For all $\sigma, \tau, \rho \in \mathbb{R}^\omega$,

$$\begin{aligned} \sigma + 0 &= \sigma \\ \sigma + \tau &= \tau + \sigma \end{aligned} \tag{3}$$

$$\sigma + (\tau + \rho) = (\sigma + \tau) + \rho \tag{4}$$

Here is a first very easy exercise in coinductive reasoning:

Proof of (3): Let $R = \{ \langle \sigma + \tau, \tau + \sigma \rangle \mid \sigma, \tau \in \mathbb{R}^\omega \}$. Since $(\sigma + \tau)(0) = \sigma(0) + \tau(0) = (\tau + \sigma)(0)$ and

$$\begin{aligned} (\sigma + \tau)' &= (\sigma' + \tau') \\ R \quad (\tau' + \sigma') \\ &= (\tau + \sigma)' \end{aligned}$$

for any σ and τ , R is a bisimulation relation on $\mathbb{R}^\omega$. The identity now follows by coinduction (Theorem 2.1). $\square$

For the motivation of the next operator, let us briefly look at the world of (real-valued) functions, which at more than one occasion, will be a source of inspiration for stream calculus. Consider a function $f : \mathbb{R} \rightarrow \mathbb{R}$ and a stream (of coefficients) $\sigma = (s_0, s_1, s_2, \dots)$ such that $f(x) = s_0 + s_1x + s_2x^2 + \dots$. Writing $\bar{f}(x) = s_1 + s_2x + s_3x^2 + \dots$, we have

$$f(x) = s_0 + (x \times \bar{f}(x))$$

Let $g : \mathbb{R} \rightarrow \mathbb{R}$ and $\tau = (t_0, t_1, t_2, \dots)$ be another such function and stream, with $g(x) = t_0 + t_1x + t_2x^2 + \dots$ and write, similarly, $g(x) = t_0 + (x \times \bar{g}(x))$. Computing now the (elementwise) function product $f(x) \times g(x)$, one finds

$$\begin{aligned} f(x) \times g(x) &= (s_0 + (x \times \bar{f}(x))) \times (t_0 + (x \times \bar{g}(x))) \\ &= (s_0 \times t_0) + x \times (\bar{f}(x) \times g(x) + s_0 \times \bar{g}(x)) \end{aligned}$$

This is one way of motivating the following definition. (An alternative, equally valid way would be to view streams as sets of words, that is, languages, with multiplicities, and stream concatenation as language concatenation. More about this later.) Let the *(convolution) product* $\sigma \times \tau$ of two streams σ and τ be the unique stream satisfying:

behavioural differential equation	initial value	name
$(\sigma \times \tau)' = (\sigma' \times \tau) + (\sigma(0) \times \tau')$	$(\sigma \times \tau)(0) = \sigma(0) \times \tau(0)$	(convolution) product

(Note that we are using the same symbol $\times$ both for the product of streams and the product of real numbers. Further note that in the above definition, $\sigma(0) \times \tau'$ is a shorthand for $[\sigma(0)] \times \tau'$.) We shall use the following standard conventions: for all $n \geq 0$,

$$-\sigma \equiv [-1] \times \sigma, \quad \sigma\tau \equiv \sigma \times \tau, \quad \sigma^0 \equiv 1, \quad \sigma^{n+1} \equiv \sigma \times \sigma^n$$

On the basis of the analogy between function multiplication and stream multiplication, one might have expected

$$(\sigma \times \tau)' = (\sigma' \times \tau) + (\sigma \times \tau')$$

which generally is *not* valid, indeed. Apparently, stream differentiation does not quite correspond to function derivation in analysis. Rather, it corresponds to the above transformation of a function f into $\bar{f}$, an operation which is not usually present in analysis. Later we shall see variations on both the notion of stream derivative and the operator of multiplication, which have more familiar properties. Let it be noted, however, that in stream calculus, both the operations of stream derivative and of convolution product, with their non-standard properties, are of central importance.

There is also the following formula for the n -th element of $\sigma \times \tau$, for any $n \geq 0$,

$$(\sigma \times \tau)(n) = \sum_{k=0}^n \sigma(n-k) \times \tau(k) \tag{5}$$

which could have been (and traditionally is) taken as an alternative definition of $\sigma \times \tau$. (Since it will play no role in what follows, a proof of this identity, which would not be too difficult, is omitted.) Note that stream product shares this property with the function product considered above, since

$$f(x) \times g(x) = s_0t_0 + (s_1t_0 + s_0t_1) \times x + (s_2t_0 + s_1t_1 + s_0t_2) \times x^2 + \dots$$

As we shall see time and again, coinductive reasoning directly in terms of the above behavioural differential equation is quite a bit simpler than the use of (5), because of the summation over the indices k and $(n-k)$ in the latter. (Equally importantly, we shall see examples of other definitions, such as that of inverse below, where no formula for the n -th element is known.)

Having constants, sum, and product at our disposal, we are ready to formulate the first theorem of stream calculus. First we state the following basic properties

$$\begin{aligned} 0 \times \sigma &= 0 \\ 1 \times \sigma &= \sigma \\ [r] \times [s] &= [r \times s] \end{aligned}$$

which are immediate by rather trivial coinduction.

Theorem 4.1 [Fundamental Theorem] *For all streams $\sigma \in \mathbb{R}^\omega$: $\sigma = \sigma(0) + (X \times \sigma')$.*

The name ‘Fundamental Theorem’ is chosen in analogy to analysis. Viewing left multiplication with the constant stream X as a kind of stream integration, the theorem tells us that stream derivation and stream integration are inverse operations: the equality gives a way of obtaining σ out of σ' (and the initial value $\sigma(0)$). As a consequence, the Fundamental Theorem enables us to solve differential equations in stream calculus in an algebraic manner, which will be the subject of Section 5.

Proof of Theorem 4.1: Define $R = \{\langle \sigma, \sigma \rangle \mid \sigma \in \mathbb{R}^\omega\} \cup \{\langle \sigma, \sigma(0) + (X \times \sigma') \rangle \mid \sigma \in \mathbb{R}^\omega\}$. We have

$$\begin{aligned} \sigma' & R \sigma' \\ &= (\sigma(0) + (X \times \sigma'))' \end{aligned}$$

from which it follows that R is a bisimulation (all other conditions that are to be checked are trivial). The theorem follows therefore by coinduction. $\square$

The proof of the following basic property of stream product introduces a generalisation of the coinduction proof principle that turns out to be extremely useful:

$$\sigma \times (\tau + \rho) = (\sigma \times \tau) + (\sigma \times \rho) \quad (6)$$

Proof of (6): Let $Q = \{\langle \sigma \times (\tau + \rho), (\sigma \times \tau) + (\sigma \times \rho) \rangle \mid \sigma, \rho, \tau \in \mathbb{R}^\omega\}$. The initial values of the first and second components of such pairs are clearly the same. Computing derivatives, we find

$$\begin{aligned} (\sigma \times (\tau + \rho))' &= (\sigma' \times (\tau + \rho)) + (\sigma(0) \times (\tau' + \rho')) \\ Q + Q &= ((\sigma' \times \tau) + (\sigma' \times \rho)) + ((\sigma(0) \times \tau') + (\sigma(0) \times \rho')) \\ &= ((\sigma' \times \tau) + (\sigma(0) \times \tau')) + ((\sigma' \times \rho) + (\sigma(0) \times \rho')) \quad [\text{using (3) and (4)}] \\ &= ((\sigma \times \tau) + (\sigma \times \rho))' \end{aligned}$$

where $Q + Q$ has the obvious meaning: for all $\alpha, \beta, \gamma, \delta \in \mathbb{R}^\omega$,

$$(\alpha + \beta) Q + Q (\gamma + \delta) \text{ iff } (\alpha Q \gamma) \text{ and } (\beta Q \delta)$$

We see that the derivatives themselves are not related by Q , but each consists of a sum of streams that are pairwise related. This type of relation is an instance of the following generalisation of the notion of bisimulation.

A relation $R \subseteq \mathbb{R}^\omega \times \mathbb{R}^\omega$ is a *bisimulation-up-to* if, for all $\sigma, \tau \in \mathbb{R}^\omega$: if $\sigma R \tau$ then

1. $\sigma(0) = \tau(0)$
2. There exist $n \geq 0$, $\alpha_0, \dots, \alpha_n, \beta_0, \dots, \beta_n \in \mathbb{R}^\omega$, such that $\sigma' = \alpha_0 + \dots + \alpha_n$ and $\tau' = \beta_0 + \dots + \beta_n$ and, for all $0 \leq i \leq n$: either $\alpha_i = \beta_i$ or $\alpha_i R \beta_i$.

The latter condition will usually be denoted by

$$\alpha_0 + \dots + \alpha_n (\Sigma R) \beta_0 + \dots + \beta_n$$

thus generalising our notation $Q + Q$ from above.

Theorem 4.2 [Coinduction-up-to] *For all $\sigma, \tau \in \mathbb{R}^\omega$:*

$$(\exists R, \text{ bisimulation-up-to: } \sigma R \tau) \Rightarrow \sigma = \tau$$

Proof: Let $R \subseteq \mathbb{R}^\omega \times \mathbb{R}^\omega$ be a bisimulation-up-to with $\sigma R \tau$. Let $\bar{R}$ be the smallest relation satisfying

1. $R \subseteq \bar{R}$
2. $\{\langle \sigma, \sigma \rangle \mid \sigma \in \mathbb{R}^\omega\} \subseteq \bar{R}$
3. for all $\alpha_0, \alpha_1, \beta_0, \beta_1 \in \mathbb{R}^\omega$, if $\alpha_0 \bar{R} \beta_0$ and $\alpha_1 \bar{R} \beta_1$ then $(\alpha_0 + \alpha_1) \bar{R} (\beta_0 + \beta_1)$.

Using the fact that R is a bisimulation-up-to, one can easily prove by induction on the definition of $\bar{R}$ that the latter is an (ordinary) bisimulation relation on $\mathbb{R}^\omega$. Since $R \subseteq \bar{R}$, the theorem follows by (ordinary) coinduction, Theorem 2.1. $\square$

Proof of (6), continued: Clearly the relation Q introduced above is a bisimulation-up-to. The identity now follows by coinduction-up-to. $\square$

Similarly, one proves

$$(\tau + \rho) \times \sigma = (\tau \times \sigma) + (\rho \times \sigma) \quad (7)$$

Stream product is also associative:

$$\sigma \times (\tau \times \rho) = (\sigma \times \tau) \times \rho \quad (8)$$

Proof of (8): Define $R = \{(\sigma \times (\tau \times \rho), (\sigma \times \tau) \times \rho) \mid \sigma, \tau, \rho \in \mathbb{R}^\omega\}$. Initial values are okay and computing derivatives gives

$$\begin{aligned} & (\sigma \times (\tau \times \rho))' \\ &= \sigma' \times (\tau \times \rho) + \sigma(0) \times (\tau' \times \rho + \tau(0) \times \rho') \\ &= \sigma' \times (\tau \times \rho) + \sigma(0) \times (\tau' \times \rho) + \sigma(0) \times (\tau(0) \times \rho') \quad [\text{using (6)}] \\ \Sigma R \quad & (\sigma' \times \tau) \times \rho + (\sigma(0) \times \tau') \times \rho + (\sigma(0) \times \tau(0)) \times \rho' \\ &= (\sigma' \times \tau + \sigma(0) \times \tau') \times \rho + (\sigma(0) \times \tau(0)) \times \rho' \quad [\text{using (7)}] \\ &= ((\sigma \times \tau) \times \rho)' \end{aligned}$$

Thus R is a bisimulation-up-to, and identity (8) follows by coinduction-up-to. $\square$

Stream product is commutative:

$$\sigma \times \tau = \tau \times \sigma \quad (9)$$

even though the shape of its defining behavioural differential equation is not symmetric. (Later we shall benefit from this asymmetry, since the definition of product can be immediately generalised to streams over alphabets of many variables. For such streams, product will no longer be commutative.) Identity (9) is an easy consequence of (5), but let us try to resist the temptation offered by the use of explicit formulae for the elements of streams, and present a proof by coinduction.

Proof of (9): The following identities will be used as lemmata in the proof. For all $r \in \mathbb{R}$, $\sigma \in \mathbb{R}^\omega$,

$$r \times \sigma = \sigma \times r \quad (10)$$

$$(X \times \sigma)' = \sigma \quad (11)$$

$$\sigma \times X = X \times \sigma \quad (12)$$

The first two identities are easy. For (12), define $R = \{(\sigma \times X, X \times \sigma) \mid \sigma \in \mathbb{R}^\omega\}$. Since

$$\begin{aligned} (\sigma \times X)' &= (\sigma' \times X) + \sigma(0) \\ &= \sigma(0) + (\sigma' \times X) \\ \Sigma R \quad & \sigma(0) + (X \times \sigma') \\ &= \sigma \quad [\text{Theorem 4.1}] \\ &= (X \times \sigma)' \quad [\text{by (11)}] \end{aligned}$$

R is a bisimulation-up-to, and identity (12) now follows by coinduction-up-to. Next let $Q = \{ \langle \sigma \times \tau, \tau \times \sigma \rangle \mid \sigma, \tau \in \mathbb{R}^\omega \}$. Initial values coincide and computing derivatives gives

$$\begin{aligned}
(\sigma \times \tau)' &= ((\sigma(0) + (X \times \sigma')) \times \tau)' \quad [\text{Theorem 4.1}] \\
&= (\sigma(0) \times \tau)' + (X \times \sigma' \times \tau)' \\
&= (\sigma(0) \times \tau)' + (\sigma' \times \tau) \quad [\text{identity (11)}] \\
\Sigma R \quad &(\sigma(0) \times \tau)' + (\tau \times \sigma') \\
&= (\tau \times \sigma(0))' + (X \times (\tau \times \sigma'))' \quad [\text{identities (10) and (11)}] \\
&= (\tau \times \sigma(0))' + (\tau \times (X \times \sigma'))' \quad [\text{identity (12)}] \\
&= (\tau \times (\sigma(0) + (X \times \sigma')))' \\
&= (\tau \times \sigma)' \quad [\text{Theorem 4.1}]
\end{aligned}$$

This proves that Q is a bisimulation-up-to and (9) follows by coinduction-up-to. $\square$

As was already suggested by the comparison between function product and stream product, streams can be viewed as power series in one formal variable (namely, the constant stream X). In order to make this more precise, we need to introduce generalised sums of streams. Countable ones will be sufficient for our purposes. Let therefore $\sigma_0, \sigma_1, \sigma_2, \dots$ be a sequence of streams such that, for all $k \geq 0$, $\sigma_0(k) + \sigma_1(k) + \sigma_2(k) + \dots < \infty$. Such a sequence is called *summable*. We define:

behavioural differential equation	initial value	name
$(\sum_{n=0}^{\infty} \sigma_n)' = \sum_{n=0}^{\infty} (\sigma_n)'$	$(\sum_{n=0}^{\infty} \sigma_n)(0) = \sum_{n=0}^{\infty} \sigma_n(0)$	generalised sum

Often we shall also write

$$\sum_{n=0}^{\infty} \sigma_n = \sigma_0 + \sigma_1 + \sigma_2 + \dots$$

Infinite sum is as well-behaved as binary sum. For instance, we have

$$\left(\sum_{n=0}^{\infty} \sigma_n \right) \times \tau = \sum_{n=0}^{\infty} (\sigma_n \times \tau)$$

and similarly for the other basic properties of sum. For the proof of the theorem below, we shall need the following basic property, which can be easily proved by induction: for all $n \geq 0$,

$$(X^{n+1})' = X^n \tag{13}$$

Theorem 4.3 *For all streams $\sigma = (s_0, s_1, s_2, \dots)$, the sequence of streams*

$$s_0, s_1 \times X^1, s_2 \times X^2, \dots$$

is summable, and satisfies

$$\sigma = s_0 + (s_1 \times X^1) + (s_2 \times X^2) + \dots$$

Proof: Using the defining equation for infinite sum; the fact that if a stream σ is summable, then also σ' is summable; and identity (13), the proof is a straightforward coinduction. $\square$

Streams for which the power series on the right is finite, are called *polynomial*. That is, a stream π is polynomial if there exists $n \geq 0$ and real numbers $p_0, p_1, \dots, p_n$ such that

$$\pi = p_0 + p_1 X + \dots + p_n X^n = (p_0, p_1, \dots, p_n, 0, 0, 0, \dots)$$

For instance,

$$\begin{aligned} 1 - X &= (1, -1, 0, 0, 0, \dots) \\ 1 + 7X + \sqrt{2}X^5 &= (1, 7, 0, 0, 0, \sqrt{2}, 0, 0, 0, \dots) \end{aligned}$$

both are polynomial.

Before we show some applications of Theorem 4.3 (identities (18)-(24) below), we first introduce an operator on streams that is *inverse* to multiplication in the following sense. Given a stream σ , we look for a stream σ^{-1} such that $\sigma \times \sigma^{-1} = 1$. Constant streams $[r] = (r, 0, 0, 0, \dots)$ with $r \neq 0$, clearly have $[r]^{-1} = (r^{-1}, 0, 0, 0, \dots) = [r^{-1}]$ as their inverse. Considering now an arbitrary stream σ with $\sigma(0) \neq 0$, and assuming for a second that σ^{-1} indeed existed, it would have to satisfy

$$0 = 1' = (\sigma \times \sigma^{-1})' = \sigma' \times \sigma^{-1} + \sigma(0) \times (\sigma^{-1})'$$

Multiplying (on the left) with the inverse of (the stream) $\sigma(0)$, one obtains

$$(\sigma^{-1})' = -\sigma(0)^{-1} \times (\sigma' \times \sigma^{-1})$$

Thus the equation $1 = \sigma \times \sigma^{-1}$ determines what $(\sigma^{-1})'$ should be. It also determines the initial value $(\sigma^{-1})(0) = \sigma(0)^{-1}$. What has happened is that we have derived from our ‘specification’ $\sigma \times \sigma^{-1} = 1$ a behavioural differential equation, which we can now simply take as the definition of inverse. For all $\sigma \in \mathbb{R}^\omega$ with $\sigma \neq 0$, let σ^{-1} be the unique stream satisfying the following equation:

behavioural differential equation	initial value	name
$(\sigma^{-1})' = -\sigma(0)^{-1} \times (\sigma' \times \sigma^{-1})$	$(\sigma^{-1})(0) = \sigma(0)^{-1}$	(multiplicative) inverse

Whenever writing σ^{-1} , we shall silently assume that $\sigma(0) \neq 0$. As usual, we shall write

$$\sigma^{-n} \equiv (\sigma^{-1})^n, \quad \frac{\sigma}{\tau} \equiv \sigma \times \tau^{-1}$$

With this convention, we can write $(\sigma^{-1})'$ as

$$\left(\frac{1}{\sigma}\right)' = \frac{-\sigma'}{\sigma(0) \times \sigma}$$

which is a bit easier to remember. (For similar reasons as with stream multiplication, stream derivation of inverse behaves differently from what we are used to in analysis. In particular, $(\sigma^{-1})' = -\sigma' \times \sigma^{-2}$ generally does *not* hold.)

Was our preference to use behavioural differential equations for the definition of the operators so far based on either obsession (the constants) or computational convenience (product), in the case of inverse, it turns out to be essential. For inverse, no explicit formula such as (5) for product, is known (not in the literature, not to us):

$$\left(\frac{1}{\sigma}\right)(n) = ?$$

(The best one can do is to define inverse by means of a recurrence relation as follows. The elements of σ^{-1} are given by $\sigma^{-1}(0) = \sigma(0)^{-1}$ and

$$\sigma^{-1}(n) = \sigma(0)^{-1} \sum_{k=0}^{n-1} \sigma(n-k) \times \sigma^{-1}(k)$$

for $n \geq 1$. This is an immediate consequence of the requirement that $\sigma \times \sigma^{-1} = 1$ and identity (5) above. Note that this recurrence is of so-called unbounded order: the n -th element $\sigma^{-1}(n)$ depends on all of the n preceding values $\sigma^{-1}(0)$ through $\sigma^{-1}(n-1)$. Reasoning in terms of such recurrences is not only extremely tedious but even next to impossible.)

The good news is that in stream calculus, there is no need for a closed formula for $\sigma^{-1}(n)$. All our reasoning about inverse, in fact about all streams and stream operators, will be coinductive,

calculating with stream derivatives as specified by behavioural differential equations. For instance, the following properties are readily verified by coinduction: for all $\sigma, \tau \in \mathbb{R}^\omega$,

$$\sigma \times \sigma^{-1} = 1 \quad (14)$$

$$\sigma^{-1} \times \sigma = 1 \quad (15)$$

$$(\sigma^{-1})^{-1} = \sigma \quad (16)$$

$$(\sigma \times \tau)^{-1} = \tau^{-1} \times \sigma^{-1} \quad (17)$$

Proof of (14)-(17): For identity (14) note that

$$\begin{aligned} (\sigma \times \sigma^{-1})' &= (\sigma' \times \sigma^{-1}) + (\sigma(0) \times (-\sigma(0)^{-1} \times \sigma' \times \sigma^{-1})) \\ &= (\sigma' \times \sigma^{-1}) + (-\sigma' \times \sigma^{-1}) \\ &= 0 \end{aligned}$$

and coinduction does the rest. Identity (15) clearly follows from (14) and the commutativity of convolution product (9). A direct coinductive proof not using the latter is also possible and is interesting in itself, since it will also apply to the inverse operator in the multivariate case in Section 15, where product will no longer be commutative. So we let

$$R = \{ \langle \rho \times (\sigma^{-1} \times \sigma), \rho \rangle \mid \rho, \sigma \in \mathbb{R}^\omega \}$$

and calculate as follows to see that R is a bisimulation-up-to:

$$\begin{aligned} (\rho \times (\sigma^{-1} \times \sigma))' &= \rho' \times (\sigma^{-1} \times \sigma) + \\ &\quad \rho(0) \times ((-\sigma(0)^{-1} \times \sigma' \times \sigma^{-1}) \times \sigma + (\sigma(0)^{-1} \times \sigma')) \\ &= \rho' \times (\sigma^{-1} \times \sigma) - \\ &\quad \rho(0) \times \sigma(0)^{-1} \times \sigma' \times (\sigma^{-1} \times \sigma) + \\ &\quad \rho(0) \times \sigma(0)^{-1} \times \sigma' \\ \Sigma R \quad \rho' - (\rho(0) \times \sigma(0)^{-1} \times \sigma') + (\rho(0) \times \sigma(0)^{-1} \times \sigma') \\ &= \rho' \end{aligned}$$

Identity (15) then follows by coinduction-up-to. For (16), note that

$$\begin{aligned} ((\sigma^{-1})^{-1})' &= -(\sigma^{-1}(0))^{-1} \times (\sigma^{-1})' \times (\sigma^{-1})^{-1} \\ &= -\sigma(0) \times (-\sigma(0)^{-1} \times \sigma' \times \sigma^{-1}) \times (\sigma^{-1})^{-1} \\ &= \sigma' \times (\sigma^{-1} \times (\sigma^{-1})^{-1}) \\ &= \sigma' \quad [\text{by identity (14)}] \end{aligned}$$

and use coinduction. Identity (17) follows from

$$\begin{aligned} (\sigma \times \tau) \times (\tau^{-1} \times \sigma^{-1}) &= \sigma \times (\tau \times \tau^{-1}) \times \sigma^{-1} \\ &= \sigma \times \sigma^{-1} \\ &= 1 \end{aligned}$$

and the observation, by (14), that inverse is uniquely defined. $\square$

Here are a few examples of streams involving the operation of inverse. The stream $(1 - X)^{-1}$ is particularly well-behaved in that it equals its own derivative:

$$\left(\frac{1}{1 - X} \right)' = \frac{1}{1 - X}$$

(We shall elaborate extensively on this fact later.) As an immediate consequence, one has

$$\frac{1}{1 - X} = 1 + X + X^2 + X^3 + \dots \quad (18)$$

or, in a notation that emphasizes the fact that we are dealing with streams,

$$\frac{1}{(1, -1, 0, 0, 0, \dots)} = (1, 1, 1, \dots)$$

Here are some further identities:

$$X \times (s_0, s_1, s_2, \dots) = (0, s_0, s_1, s_2, \dots) \quad (19)$$

$$\frac{1}{1+X} = 1 - X + X^2 - X^3 + \dots \quad (20)$$

$$\frac{1}{1-X^2} = 1 + X^2 + X^4 + \dots \quad (21)$$

$$\frac{1}{(1-X)^2} = 1 + 2X + 3X^2 + 4X^3 \dots \quad (22)$$

$$\frac{1}{1-rX} = 1 + rX + r^2X^2 + r^3X^3 + \dots \quad (23)$$

$$\frac{X}{1+X^2} = X - X^3 + X^5 - X^7 + \dots \quad (24)$$

Proof of (19)-(24): For instance, (20) follows by coinduction from the fact that

$$\{ \langle (1+X)^{-1}, 1 - X + X^2 - X^3 + \dots \rangle, \langle -(1+X)^{-1}, -1 + X - X^2 + X^3 - \dots \rangle \}$$

is a bisimulation relation on $\mathbb{R}^\omega$. Similarly, (22) follows by coinduction since

$$\{ \langle \frac{1}{(1-X)^2} + \frac{k}{1-X}, (k+1) + (k+2)X + (k+3)X^2 + \dots \rangle \mid k \geq 0 \}$$

is a bisimulation relation. □

The definition of the following operation on streams is again clearly inspired by analysis. For all streams σ and τ with $\tau(0) = 0$, the *composition* $\sigma \circ \tau$ is defined as the unique stream satisfying the following behavioural differential equation:

behavioural differential equation	initial value
$(\sigma \circ \tau)' = \tau' \times (\sigma' \circ \tau)$	$(\sigma \circ \tau)(0) = \sigma(0)$

We shall often write

$$\sigma \circ \tau \equiv \sigma(\tau)$$

The condition that $\tau(0) = 0$ is needed in the proof of the following identity, which shows that composition behaves as usual. For $\sigma = (s_0, s_1, s_2, \dots)$ and τ with $\tau(0) = 0$,

$$\sigma(\tau) = s_0 + (s_1 \times \tau^1) + (s_2 \times \tau^2) + \dots \quad (25)$$

Proof of (25): Because $\tau(0) = 0$, we have $(\tau^{n+1})' = \tau' \times \tau^n$, for all $n \geq 0$. As a consequence,

$$\begin{aligned} & (s_0 + (s_1 \times \tau^1) + (s_2 \times \tau^2) + \dots)' \\ &= (s_1 \times \tau') + (s_2 \times \tau' \times \tau) + (s_3 \times \tau' \times \tau^2) + \dots \\ &= \tau' \times (s_1 + (s_2 \times \tau) + (s_3 \times \tau^2) + \dots) \end{aligned}$$

Now (with $\sigma = (s_0, s_1, s_2, \dots)$) let

$$R = \{ \langle \rho \times \sigma(\tau), \rho \times (s_0 + (s_1 \times \tau^1) + (s_2 \times \tau^2) + \dots) \rangle \mid \sigma, \tau, \rho \in \mathbb{R}^\omega, \tau(0) = 0 \}$$

Because

$$\begin{aligned}
(\rho \times \sigma(\tau))' &= (\rho' \times \sigma(\tau)) + (\rho(0) \times (\tau' \times \sigma'(\tau))) \\
&= (\rho' \times \sigma(\tau)) + (\rho(0) \times \tau') \times \sigma'(\tau) \\
\Sigma R \quad &\rho' \times (s_0 + (s_1 \times \tau^1) + (s_2 \times \tau^2) + \dots) + \\
&\quad (\rho(0) \times \tau') \times (s_1 + (s_2 \times \tau^1) + (s_3 \times \tau^2) + \dots) \\
&= \rho' \times (s_0 + (s_1 \times \tau^1) + (s_2 \times \tau^2) + \dots) + \\
&\quad \rho(0) \times (\tau' \times (s_1 + (s_2 \times \tau^1) + (s_3 \times \tau^2) + \dots)) \\
&= (\rho \times (s_0 + (s_1 \times \tau^1) + (s_2 \times \tau^2) + \dots))'
\end{aligned}$$

R is a bisimulation-up-to, and identity (25) follows by coinduction-up-to. $\square$

Here are a few useful instances and applications of this fact:

$$\begin{aligned}
\sigma(X) &= \sigma \\
\sigma(-X) &= s_0 - s_1 X + s_2 X^2 - s_3 X^3 + \dots \\
&= (s_0, -s_1, s_2, -s_3, \dots) \\
\frac{1}{2}(\sigma(X) + \sigma(-X)) &= (s_0, 0, s_3, 0, \dots) \\
\sigma(X^2) &= s_0 + s_1 X^2 + s_2 X^4 + s_3 X^6 + \dots
\end{aligned} \tag{26}$$

5 Solving behavioural differential equations

The calculus we have just developed will now be used to solve a number of behavioural differential equations in an algebraic manner. In particular, it will be shown how to solve (homogeneous or non-homogeneous) linear equations, possibly with non-constant coefficients. In Sections 6 and 12, these solutions will be used to solve difference equations and analytical differential equations of a similar kind.

Let us start with linear behavioural differential equations with constant coefficients. The method we are about to explain will apply to any equation of the following type:

behavioural differential equation	initial value
$\sigma^{(n)} + r_{n-1}\sigma^{(n-1)} + \dots + r_1\sigma^{(1)} + r_0\sigma = \tau$	$\sigma^{(k)}(0) = c_k, \quad 0 \leq k \leq n-1$

where $n \geq 1$, $\sigma^{(k)}$ denotes the k -th stream derivative of σ , $r_0, \dots, r_{n-1}$ are real numbers (interpreted as constant streams $[r_k]$), $c_0, \dots, c_{n-1}$ are also real numbers, and τ is an arbitrary stream. If $\tau = 0$ the equation is called homogeneous, otherwise it is called non-homogeneous. The claim is that

1. there exists a unique stream σ satisfying the equation;
2. σ can be expressed in terms of τ , the coefficients $r_0, \dots, r_{n-1}$, the initial values $c_0, \dots, c_{n-1}$, and the operators of plus, product and inverse.

Rather than giving a proof of this claim in its full generality, we prefer to treat a few special cases, being (lazy and) confident that they will bring the main idea across. Let us start with a simple example of a homogeneous equation:

behavioural differential equation	initial value
$\sigma'' - \sigma' - \sigma = 0$	$\sigma(0) = 0, \sigma'(0) = 1$

In order to solve it we multiply the left and right side of the equation with X^2 (where 2 is the coefficient of the highest derivative):

$$(X^2 \times \sigma'') - (X^2 \times \sigma') - (X^2 \times \sigma) = 0$$

Applying the Fundamental Theorem 4.1 to both σ and σ' , we obtain, using $\sigma(0) = 0$ and $\sigma'(0) = 1$,

$$\sigma = X \times \sigma', \quad \sigma' = 1 + (X \times \sigma'')$$

implying $X^2 \times \sigma'' = -X + \sigma$ and $X^2 \times \sigma' = X \times \sigma$. Substituting this above gives

$$\begin{aligned} 0 &= (X^2 \times \sigma'') - (X^2 \times \sigma') - (X^2 \times \sigma) \\ &= -X + \sigma - (X \times \sigma) - (X^2 \times \sigma) \\ &= -X + (1 - X - X^2) \times \sigma \end{aligned}$$

yielding as the final outcome the following expression for the solution of the differential equation:

$$\sigma = \frac{X}{(1 - X - X^2)}$$

One could continue by manipulating this expression still further in order to obtain a so-called *closed formula* for the n -th element of σ , expressing $\sigma(n)$ in terms of a formula on the natural numbers depending on (the variable) n . We shall come back to this in Section 6.

Here is another example, this time non-homogeneous:

behavioural differential equation	initial value
$\sigma' - \sigma = (1 - X)^{-1}$	$\sigma(0) = 1$

By Theorem 4.1, we obtain $X \times \sigma' = \sigma - 1$, whence

$$\begin{aligned} X \times (1 - X)^{-1} &= (X \times \sigma') - (X \times \sigma) \\ &= \sigma - 1 - (X \times \sigma) \\ &= -1 + (1 - X) \times \sigma \end{aligned}$$

As a consequence, the following expression is obtained:

$$\sigma = \frac{1}{(1 - X)^2} \tag{27}$$

The following table contains some further examples which the reader may wish to use to test his or her stream-calculus abilities:

behavioural differential equation	initial value
$\sigma' - 5\sigma = -2(1 - 2X)^{-1}$	$\sigma(0) = 3$
$\tau'' + r^2\tau = 1$	$\tau(0) = 1, \tau'(0) = 0$
$\rho'' - \rho' = 2 + 6X$	$\rho(0) = 0, \rho'(0) = 0$

The solutions of these equations are

$$\sigma = \frac{3 - 8X}{1 - 7X + 10X^2}, \quad \tau = \frac{1 + X + X^2}{1 + r^2X^2}, \quad \rho = \frac{1 - X + 2X + 6X^3}{1 - X}$$

The fact that the coefficients of the equations so far have been constant real numbers is by no means crucial. Here is the more general formulation:

behavioural differential equation	initial value
$\sigma^{(n)} + (\rho_{n-1} \times \sigma^{(n-1)}) + \dots + (\rho_1 \times \sigma^{(1)}) + (\rho_0 \times \sigma) = \tau$	$\sigma^{(k)}(0) = c_k, \quad 0 \leq k \leq n - 1$

where now not only τ but also $\rho_0, \dots, \rho_{n-1}$ are arbitrary streams. The claim is again that

1. there exists a unique stream σ satisfying the equation;
2. σ can be expressed in terms of the streams τ and $\rho_0, \dots, \rho_{n-1}$, the initial values $c_0, \dots, c_{n-1}$, and the operators of plus, product and inverse.

The proof for arbitrary $n \geq 1$ is not more difficult (just more writing) than the proof for $n = 1$, which is presented next. Consider the equation

behavioural differential equation	initial value
$\sigma' + (\rho \times \sigma) = \tau$	$\sigma(0)$

where ρ and τ are arbitrary streams. Calculating as before (multiplying both sides of the equation by X , invoking Theorem 4.1), we obtain for the solution of this equation the following expression:

$$\sigma = \frac{\sigma(0) + (X \times \tau)}{1 + (X \times \rho)}$$

6 Application: solving difference equations

The techniques of Section 5 will next be used to solve linear difference equations (also called linear recurrence relations) with constant coefficients. (An example of an equation with non-constant coefficients will be dealt with in Section 9.) The main idea is to transform difference equations, which can be seen as inductive definitions of streams, in a canonical fashion into behavioural differential equations. The heart of the matter is thus a systematic transformation of inductive stream definitions into coinductive ones.

More precisely, we consider non-homogeneous linear difference equations of order k with constant coefficients:

difference equation	initial values
$s_{n+k} + r_{k-1}s_{n+k-1} + \dots + r_1s_{n+1} + r_0s_n = t_n$	$s_0, \dots, s_{k-1}$

where $k \geq 1$ (the order of the equation), $n \geq 0$, the coefficients $r_0, \dots, r_{k-1}$ are real numbers (the coefficient for s_{n+k} has been taken identical to 1 for convenience), and $t_0, t_1, t_2, \dots$ is an arbitrary sequence of real numbers. Defining

$$\sigma = (s_0, s_1, s_2, \dots), \quad \tau = (t_0, t_1, t_2, \dots)$$

we set out to transform the above difference equation into a behavioural differential equation in the variable σ , by multiplying both sides of the equation with the constant stream X^n :

$$s_{n+k}X^n + r_{k-1}s_{n+k-1}X^n + \dots + r_1s_{n+1}X^n + r_0s_nX^n = t_nX^n$$

Next we take on both sides the infinite sum over all $n \geq 0$:

$$\sum_{n=0}^{\infty} s_{n+k}X^n + r_{k-1} \sum_{n=0}^{\infty} s_{n+k-1}X^n + \dots + r_1 \sum_{n=0}^{\infty} s_{n+1}X^n + r_0 \sum_{n=0}^{\infty} s_nX^n = \sum_{n=0}^{\infty} t_nX^n$$

Since, for any $i \geq 0$,

$$\sigma^{(i)} = s_i + s_{i+1}X^1 + s_{i+2}X^2 + \dots$$

which is an immediate corollary of Theorem 4.3, we obtain

$$\sigma^{(k)} + r_{k-1}\sigma^{(k-1)} + \dots + r_1\sigma^{(1)} + r_0\sigma = \tau$$

with initial values

$$\sigma(0) = s_0, \quad \sigma^{(1)}(0) = s_1, \quad \dots, \quad \sigma^{(k-1)}(0) = s_{k-1}$$

And so we are done, since we have learned in Section 5 how to solve this type of behavioural differential equation.

Let us look at a few examples. Here is maybe the most famous difference equation of all:

difference equation	initial values
$s_{n+2} - s_{n+1} - s_n = 0$	$s_0 = 0, s_1 = 1$

defining the Fibonacci numbers. The method above transforms it into

behavioural differential equation	initial values
$\sigma'' - \sigma' - \sigma = 0$	$\sigma(0) = 0, \sigma'(0) = 1$

which we recognise from Section 5, where the solution was shown to be

$$\sigma = \frac{X}{(1 - X - X^2)}$$

This in principle answers the question as to what stream is defined by the difference equation we started with. A further question that is often raised is how the n -th element $\sigma(n)$ looks like, ideally as a function of n . The method of partial fractions, well-known from the theory of generating functions, also works in stream calculus. Defining

$$r_+ = \frac{1 + \sqrt{5}}{2}, \quad r_- = \frac{1 - \sqrt{5}}{2}$$

which are the roots of $1 - X - X^2$ since $1 - X - X^2 = (1 - r_+X)(1 - r_-X)$, we have

$$\begin{aligned} \sigma &= \frac{X}{(1 - X - X^2)} \\ &= \frac{1}{r_+ - r_-} \left(\frac{1}{1 - r_+X} - \frac{1}{1 - r_-X} \right) \\ &= \frac{1}{\sqrt{5}} \left(\sum_{n \geq 0}^{\infty} r_+^n X^n - \sum_{n \geq 0}^{\infty} r_-^n X^n \right) \quad [\text{by identity (23)}] \end{aligned}$$

giving the familiar answer

$$\sigma(n) = \frac{1}{\sqrt{5}} (r_+^n - r_-^n)$$

Note that the entire game is played here inside the world of stream calculus, without any reference to (generating) functions. For a second example, non-homogeneous this time, consider

difference equation	initial value
$s_{n+1} - s_n = 1$	$s_0 = 1$

Our method transforms it into

behavioural differential equation	initial value
$\sigma' - \sigma = (1 - X)^{-1}$	$\sigma(0) = 1$

(using the fact that $(1 - X)^{-1} = 1 + X + X^2 + \dots$, identity (18)). For the solution we look again at Section 5:

$$\begin{aligned} \sigma &= (1 - X)^{-2} \quad [\text{identity (27)}] \\ &= 1 + 2X + 3X^2 + 4X^3 + \dots \quad [\text{by identity (22)}] \\ &= (1, 2, 3, 4, \dots) \end{aligned}$$

which comes as no surprise. Here is yet another example of a non-homogeneous equation:

difference equation	initial values
$s_{n+1} - 5s_n = -2^{n+1}$	$s_0 = 3$

Writing $-2^{n+1} = -2 \times 2^n$ and using identity (23), the following differential equation is obtained:

behavioural differential equation	initial value
$\sigma' - 5\sigma = -2(1 - 2X)^{-1}$	$\sigma(0) = 3$

Using the method of partial fractions again, the solution found in Section 5 can be rewritten as

$$\begin{aligned}\sigma &= \frac{3 - 8X}{1 - 7X + 10X^2} \\ &= \frac{2}{3}(1 - 2X)^{-1} + \frac{7}{3}(1 - 5X)^{-1}\end{aligned}$$

yielding, again by identity (23),

$$\sigma(n) = \frac{2}{3}2^n + \frac{7}{3}5^n$$

7 Solving quadratic equations in stream calculus

The operation of the square root of a stream is introduced, and is used to solve quadratic equations in stream calculus. (Although only square root is treated, it will be obvious how to deal with variations.)

As in the case of the definition of the inverse operator, we simply state what we wish and then derive a definition that gives us precisely that. For a stream σ , we want a stream $\sqrt{\sigma}$ such that $\sqrt{\sigma} \times \sqrt{\sigma} = \sigma$. This determines the initial value as the real number $(\sqrt{\sigma})(0) = \sqrt{\sigma(0)}$ (implying that $\sigma(0)$ should be positive). Taking derivatives on both sides of our specification, we get

$$(\sqrt{\sigma}' \times \sqrt{\sigma}) + (\sqrt{\sigma(0)} \times \sqrt{\sigma}') = \sigma'$$

implying, under the assumption that $\sigma(0) \neq 0$,

$$\sqrt{\sigma}' = \frac{\sigma'}{\sqrt{\sigma(0)} + \sqrt{\sigma}}$$

And so we arrive at the following behavioural differential equation. For all streams σ with $\sigma(0) > 0$, let $\sqrt{\sigma}$ be the unique stream satisfying the following equation:

behavioural differential equation	initial value
$\sqrt{\sigma}' = \frac{\sigma'}{\sqrt{\sigma(0)} + \sqrt{\sigma}}$	$\sqrt{\sigma}(0) = \sqrt{\sigma(0)}$

It is an immediate consequence of this definition that, indeed,

$$\sqrt{\sigma} \times \sqrt{\sigma} = \sigma \tag{28}$$

The following theorem expresses a basic property of square root.

Theorem 7.1 *For all streams σ and τ with $\sigma(0) > 0$,*

$$\text{if } \tau \times \tau = \sigma \text{ then either } \tau = \sqrt{\sigma} \text{ or } \tau = -\sqrt{\sigma}$$

depending on whether $\tau(0)$ is positive or negative.

Note that the theorem is less trivial than it seems. Although a similar statement holds for real numbers s and t with $s > 0$: if $t \times t = s$ then either $t = \sqrt{s}$ or $t = -\sqrt{s}$, such a property is not valid for arbitrary *functions*. If f and g are two real-valued functions with $f > 0$ and such that $g \times g = f$, then g need not be equal to $\sqrt{f}$ or $-\sqrt{f}$.

Proof of Theorem 7.1: Because $\tau \times \tau = \sigma$ we have $\tau(0)^2 = \sigma(0)$ and thus $\tau(0)$ is either $\sqrt{\sigma(0)}$ or $-\sqrt{\sigma(0)}$. We assume in the rest of the proof that $\tau(0) = \sqrt{\sigma(0)}$, the other case being similar. Because

$$\begin{aligned}\sigma' &= (\tau \times \tau)' \\ &= (\tau' \times \tau) + (\tau(0) \times \tau') \\ &= \tau' \times (\tau(0) + \tau) \\ &= \tau' \times (\sqrt{\sigma(0)} + \tau)\end{aligned}$$

we have for τ' a similar expression as for $\sqrt{\sigma'}$:

$$\tau' = \frac{\sigma'}{\sqrt{\sigma(0)} + \tau} \quad (29)$$

It is this equality which tells us how to build a suitable bisimulation relation, which will be a bit more complicated than the ones that we have encountered so far. Let $R \subseteq \mathbb{R}^\omega \times \mathbb{R}^\omega$ be the smallest relation such that

1. $\langle \tau, \sqrt{\sigma} \rangle \in R$
2. $\{\langle \rho, \rho \rangle \mid \rho \in \mathbb{R}^\omega\} \subseteq R$
3. For all streams $\alpha_0, \alpha_1, \beta_0, \beta_1$: if $\langle \alpha_0, \beta_0 \rangle \in R$ and $\langle \alpha_1, \beta_1 \rangle \in R$ then
 - (a) $\langle \alpha_0 + \alpha_1, \beta_0 + \beta_1 \rangle \in R$
 - (b) $\langle \alpha_0 \times \alpha_1, \beta_0 \times \beta_1 \rangle \in R$
 - (c) $\langle (\alpha_0)^{-1}, (\beta_0)^{-1} \rangle \in R$

Using identity (29) above, one can now prove, with induction on the definition of R , that R is a bisimulation relation. The theorem then follows by coinduction. $\square$

Just as an entertaining aside¹, note that the defining equation for square root immediately gives rise to a representation as a continued fraction:

$$\begin{aligned} \sqrt{\sigma} &= \sqrt{\sigma(0)} + (X \times \sqrt{\sigma'}) \quad [\text{Theorem 4.1}] \\ &= \sqrt{\sigma(0)} + \frac{X \times \sigma'}{\sqrt{\sigma(0)} + \sqrt{\sigma}} \\ &= \sqrt{\sigma(0)} + \frac{X \times \sigma'}{2\sqrt{\sigma(0)} + \frac{X \times \sigma'}{2\sqrt{\sigma(0)} + \frac{X \times \sigma'}{2\sqrt{\sigma(0)} + \frac{X \times \sigma'}{\ddots}}}} \end{aligned}$$

(We shall come back to continued fractions in Section 17.) Here are a few basic properties and examples of square roots:

$$\sqrt{\sigma} \times \sqrt{\tau} = \sqrt{\sigma \times \tau} \quad (30)$$

$$\sqrt{\frac{1}{\sigma}} = \frac{1}{\sqrt{\sigma}} \quad (31)$$

$$\sqrt{(1, 1, 1, \dots)} = (1, \frac{1}{2}, \frac{3}{8}, \frac{5}{16}, \dots) \quad (32)$$

$$\sqrt{(1, 2, 3, \dots)} = (1, 1, 1, \dots) \quad (33)$$

$$\sqrt{(1, -4, 0, 0, 0, \dots)} = (1, -2, -2, -4, -10, -28, -84, -264, -858, \dots) \quad (34)$$

Proof of identities (30)-(34): The first identity is immediate by Theorem 7.1 and the observation that

$$(\sqrt{\sigma} \times \sqrt{\tau}) \times (\sqrt{\sigma} \times \sqrt{\tau}) = \sigma \times \tau$$

¹This was suggested to us by Alexandru Baltag.

For identity (31), we calculate as follows:

$$\begin{aligned}
\sqrt{\frac{1}{\sigma}} &= 1 \times \sqrt{\frac{1}{\sigma}} \\
&= \left(\frac{1}{\sqrt{\sigma}} \times \sqrt{\sigma}\right) \times \sqrt{\frac{1}{\sigma}} \\
&= \frac{1}{\sqrt{\sigma}} \times \sqrt{\sigma \times \frac{1}{\sigma}} \quad [\text{by identity (30)}] \\
&= \frac{1}{\sqrt{\sigma}} \times \sqrt{1} \\
&= \frac{1}{\sqrt{\sigma}}
\end{aligned}$$

Example (33) follows from

$$\begin{aligned}
(1, 1, 1, \dots)^2 &= \left(\frac{1}{1-X}\right)^2 \quad [\text{by identity (18)}] \\
&= \frac{1}{(1-X)^2} \\
&= (1, 2, 3, \dots) \quad [\text{by identity (22)}]
\end{aligned}$$

The elements of the other two examples can be computed one by one, by unfolding the defining equation of the operation of square root. Alternatively, example (34) is best understood in the context of the solution of the quadratic equation below. $\square$

As an example of the use of the square root operator, we set out to solve the following quadratic equation:

$$\gamma = 1 + (X \times \gamma^2) \tag{35}$$

A first observation is that a stream γ satisfies this equation iff it satisfies the following differential equation:

behavioural differential equation	initial value
$\gamma' = \gamma \times \gamma$	$\gamma(0) = 1$

As a consequence, we note that equation (35) has a unique solution. Computing the respective stream derivatives of γ , one finds

$$\gamma = (1, 1, 2, 5, 14, \dots)$$

which the reader may recognise as the stream of so-called Catalan numbers. In order to express the stream γ in terms of constants and stream operators, now also including the square root operator, one calculates as follows:

$$\begin{aligned}
\gamma &= 1 + (X \times \gamma^2) \\
&\text{iff } (4X^2 \times \gamma^2) - (4X \times \gamma) + 4X = 0 \\
&\text{iff } (4X^2 \times \gamma^2) - (4X \times \gamma) + 1 = 1 - 4X \\
&\text{iff } (2X \times \gamma - 1)^2 = 1 - 4X \\
&\text{iff } (2X \times \gamma) - 1 = -\sqrt{1 - 4X} \quad [\text{by Theorem 7.1; note that } ((2X \times \gamma) - 1)(0) = -1]
\end{aligned}$$

Now it is tempting to conclude that

$$\gamma = \frac{1 - \sqrt{1 - 4X}}{2X}$$

but let us not get carried away: we are living in stream calculus here, and are not dealing with functions. Dividing by (the constant stream) $X = (0, 1, 0, 0, 0, \dots)$ has no meaning because inverse is only defined for streams with initial value different from 0. Fortunately, we are still able to get rid of the ' $X \times$ ', simply by stream differentiation, since $(X \times \gamma)' = \gamma$. Thus

$$\begin{aligned} \gamma &= 1 + (X \times \gamma^2) \\ \text{iff } (2X \times \gamma) - 1 &= -\sqrt{1 - 4X} \\ \text{iff } \gamma &= 1/2(1 - \sqrt{1 - 4X})' \\ \text{iff } \gamma &= \frac{2}{1 + \sqrt{1 - 4X}} \quad [\text{using the defining equation of square root}] \end{aligned}$$

It is not too difficult to generalise this example to more general quadratic equations, in one unknown σ , with as coefficients arbitrary streams α , β , and γ :

$$(\alpha \times \sigma^2) + (\beta \times \sigma) + \gamma = 0$$

and to determine the conditions on the coefficients which guarantee the existence of a unique solution.

8 Shuffle product and shuffle inverse

The power of stream calculus is further increased by the introduction of a number of new operators, notably shuffle product and shuffle inverse. They will play a role in various applications later, including the solution of analytical differential equations.

As we observed earlier, stream differentiation of (convolution) product and inverse does not behave the way we are used to from analysis. A somewhat formalistic way of motivating the definition of the following two operators is that they constitute, in this respect, more familiarly behaved alternatives to product and inverse. For streams σ and τ , let the shuffle product $\sigma \otimes \tau$ and the shuffle inverse $\sigma^{-\underline{1}}$ be the unique streams satisfying the following behavioural differential equations:

behavioural differential equation	initial value	name
$(\sigma \otimes \tau)' = (\sigma' \otimes \tau) + (\sigma \otimes \tau')$	$(\sigma \otimes \tau)(0) = \sigma(0) \times \tau(0)$	shuffle product
$(\sigma^{-\underline{1}})' = -\sigma' \otimes (\sigma^{-\underline{1}} \otimes \sigma^{-\underline{1}})$	$\sigma^{-\underline{1}}(0) = \sigma(0)^{-1}$	shuffle inverse

(Note that we use an underlined symbol $\underline{1}$ to distinguish shuffle inverse from the inverse to the convolution product.) Viewing streams again as sets of words with multiplicities in the reals, shuffle product can also be interpreted as (a generalisation of) the shuffle of languages, which provides another type of motivation and at the same time accounts for the terminology (more about this later). Yet another way of explaining the relevance of shuffle product is provided by the following property. For streams $\sigma = (s_0, s_1, s_2, \dots)$ and $\tau = (t_0, t_1, t_2, \dots)$, one has

$$\begin{aligned} & (s_0 + \frac{s_1}{1!}X^1 + \frac{s_2}{2!}X^2 + \dots) \times (t_0 + \frac{t_1}{1!}X^1 + \frac{t_2}{2!}X^2 + \dots) \\ &= (\sigma \otimes \tau)(0) + \frac{(\sigma \otimes \tau)(1)}{1!}X^1 + \frac{(\sigma \otimes \tau)(2)}{2!}X^2 + \dots \end{aligned} \tag{36}$$

which is an immediate consequence of the following formula for $(\sigma \otimes \tau)(n)$, for any $n \geq 0$:

$$(\sigma \otimes \tau)(n) = \sum_{k=0}^n \binom{n}{k} \times \sigma(n-k) \times \tau(k) \tag{37}$$

Similar to the case of convolution product, the latter formula could also have been taken as an alternative definition to the behavioural differential equation above. Reasoning in terms of (37), however, is again unnecessarily complicated, because of the use of the indices, and the occurrence

of the binomial coefficient. And as with ordinary inverse, no similar such formula for shuffle inverse is known.

We shall use the following conventions, for all $n \geq 0$:

$$\sigma^0 \equiv 1, \quad \sigma^{\underline{n+1}} \equiv \sigma \otimes \sigma^{\underline{n}}, \quad \sigma^{-\underline{n}} \equiv (\sigma^{-1})^{\underline{n}}$$

Here are a few basic properties of shuffle product and shuffle inverse. For all streams $\sigma = (s_0, s_1, s_2, \dots)$, τ and ρ , for all $r \in \mathbb{R}$, and for all $n \geq 0$,

$$\sigma \otimes \tau = \tau \otimes \sigma \tag{38}$$

$$r \otimes \sigma = r \times \sigma \tag{39}$$

$$0 \otimes \sigma = 0 \tag{40}$$

$$1 \otimes \sigma = \sigma \tag{41}$$

$$\sigma \otimes (\tau \otimes \rho) = (\sigma \otimes \tau) \otimes \rho \tag{42}$$

$$\sigma \otimes (\tau + \rho) = (\sigma \otimes \tau) + (\sigma \otimes \rho) \tag{43}$$

$$(\sigma^{\underline{n+1}})' = n \otimes \sigma' \otimes \sigma^{\underline{n}} \tag{44}$$

$$X^{\underline{n}} = n! \times X^n \tag{45}$$

$$\sigma \otimes \sigma^{-1} = 1 \tag{46}$$

$$(\sigma^{-1})^{-1} = \sigma \tag{47}$$

$$(\sigma \otimes \tau)^{-1} = \sigma^{-1} \otimes \tau^{-1} \tag{48}$$

$$X \otimes \sigma = s_0 X^1 + 2s_1 X^2 + 3s_2 X^3 + \dots \tag{49}$$

$$= (0, s_0, 2s_1, 3s_2, \dots)$$

$$(X \otimes \sigma')' = s_1 + 2s_2 X^1 + 3s_3 X^2 + \dots \tag{50}$$

$$= (s_1, 2s_2, 3s_3, \dots)$$

$$(1 - X)^{-1} = 1 + 1!X + 2!X^2 + 3!X^3 \dots \tag{51}$$

Proof of identities (38)-(51): We treat a few examples. For (38), define

$$R = \{\langle \sigma \otimes \tau, \tau \otimes \sigma \rangle \mid \sigma, \tau \in \mathbb{R}^\omega\}$$

and observe that

$$\begin{aligned} (\sigma \otimes \tau)' &= (\sigma' \otimes \tau) + (\sigma \otimes \tau') \\ \Sigma R &= (\tau \otimes \sigma') + (\tau' \otimes \sigma) \\ &= (\tau' \otimes \sigma) + (\tau \otimes \sigma') \\ &= (\tau \otimes \sigma)' \end{aligned}$$

Identity (38) now follows by coinduction-up-to (Theorem 4.2). Identity (44) can be proven with induction on n . For (46) define

$$Q = \{\langle \rho \otimes (\sigma \otimes \sigma^{-1}), \rho \rangle \mid \rho, \sigma, \tau \in \mathbb{R}^\omega\}$$

Computing derivatives gives

$$\begin{aligned} (\rho \otimes (\sigma \otimes \sigma^{-1}))' &= \rho' \otimes (\sigma \otimes \sigma^{-1}) + \\ &\quad \rho \otimes ((\sigma' \otimes \sigma^{-1}) + (\sigma \otimes (-\sigma' \otimes \sigma^{-1} \otimes \sigma^{-1}))) \\ &= \rho' \otimes (\sigma \otimes \sigma^{-1}) + \\ &\quad \rho \otimes (\sigma' \otimes \sigma^{-1}) - (\rho \otimes (\sigma' \otimes \sigma^{-1})) \otimes (\sigma \otimes \sigma^{-1}) \\ \Sigma Q &= \rho' + \rho \otimes (\sigma' \otimes \sigma^{-1}) - \rho \otimes (\sigma' \otimes \sigma^{-1}) \\ &= \rho' \end{aligned}$$

which proves that Q is a bisimulation-up-to, allowing us again to apply coinduction-up-to. For identity (49), let

$$T = \{\langle X \otimes \sigma, s_0 X^1 + 2s_1 X^2 + 3s_2 X^3 + \dots \rangle \mid \sigma = (s_0, s_1, s_2, \dots) \in \mathbb{R}^\omega\}$$

and note that

$$\begin{aligned} (X \otimes \sigma)' &= \sigma + (X \otimes \sigma') \\ \Sigma T \quad \sigma + (s_1 X^1 + 2s_2 X^2 + 3s_3 X^3 + \dots) \\ &= (s_0 + s_1 X^1 + s_2 X^2 + \dots) + (s_1 X^1 + 2s_2 X^2 + 3s_3 X^3 + \dots) \\ &\quad [\text{by Theorem 4.3}] \\ &= s_0 + 2s_1 X^1 + 3s_2 X^2 + 4s_3 X^3 + \dots \\ &= (s_0 X + 2s_1 X^2 + 3s_2 X^3 + 4s_3 X^4 + \dots)' \end{aligned}$$

showing that T is a bisimulation-up-to. Finally for (51), observe that

$$\{\langle n!(1 - X)^{-\frac{n+1}{2}}, n! + (n+1)!X + (n+2)!X^2 + \dots \rangle \mid n \geq 0\}$$

is a bisimulation relation on $\mathbb{R}^\omega$. □

There exist obvious variations for shuffle product and inverse on some of the definitions and observations regarding convolution product and inverse. For instance, using identity (45), the following Taylor expansion theorem for streams is an immediate consequence of Theorem 4.3. For a stream $\sigma = (s_0, s_1, s_2, \dots)$,

$$\begin{aligned} \sigma &= \sum_{n=0}^{\infty} \frac{\sigma^{(n)}(0)}{n!} \times X^n \\ &= s_0 + \frac{s_1}{1!} X^1 + \frac{s_2}{2!} X^2 + \dots \end{aligned}$$

And the following behavioural differential equation defines an operation of ‘shuffle square root’:

behavioural differential equation	initial value
$(\sigma^{\frac{1}{2}})' = 1/2 \otimes \sigma' \otimes (\sigma^{\frac{1}{2}})^{-1}$	$\sigma^{\frac{1}{2}}(0) = \sqrt{\sigma(0)}$

satisfying $\sigma^{\frac{1}{2}} \otimes \sigma^{\frac{1}{2}} = \sigma$.

More important for what follows is the operation of stream exponentiation. For a stream σ let $\exp(\sigma)$ be the unique stream satisfying the following behavioural differential equation:

behavioural differential equation	initial value
$\exp(\sigma)' = \sigma' \otimes \exp(\sigma)$	$\exp(\sigma)(0) = e^{\sigma(0)}$

where $e^{\sigma(0)}$ is the analytical function e^x applied to the real number $\sigma(0)$. Exponentiation has many familiar properties that we shall formulate in a second. The for our purposes most important property is less familiar, in fact rather surprising: for all $r \in \mathbb{R}$,

$$\exp(rX) = \frac{1}{1 - rX} \tag{52}$$

Here are some further identities involving exponents:

$$\exp(\sigma) = 1 + \frac{\sigma^1}{1!} + \frac{\sigma^2}{2!} + \dots \tag{53}$$

$$\exp(\sigma) \otimes \exp(\tau) = \exp(\sigma + \tau) \tag{54}$$

$$\exp(-\sigma) = \exp(\sigma)^{-1} \tag{55}$$

The following related identities will come in handy later:

$$\frac{1}{1-rX} \otimes \frac{1}{1-sX} = \frac{1}{1-(r+s)X} \quad (56)$$

$$\left(\frac{1}{1-rX}\right)^n = \frac{1}{1-nrX} \quad (57)$$

$$((1+rX)^n)' = r(1+rX)^{n-1} + r(1+rX)^{n-2} + \dots + r(1+rX) + r \quad (58)$$

$$\left(\frac{1}{(1-rX)^n}\right)' = \frac{r}{(1-rX)^n} + \frac{r}{(1-rX)^{n-1}} + \dots + \frac{r}{1-rX} \quad (59)$$

$$X^{n+1} \times \frac{1}{1-X} = -1 - X - \dots - X^n + \frac{1}{1-X} \quad (60)$$

Proof of identities (52)-(60): Again we only treat a few examples. For (52) let $R = \{ \langle s \times \exp(rX), \frac{s}{1-rX} \rangle \mid r, s \in \mathbb{R} \}$. It is a bisimulation relation, since

$$\begin{aligned} (s \times \exp(rX))' &= (r \times s) \times \exp(rX) \\ R &\frac{r \times s}{1-rX} \\ &= \left(\frac{s}{1-rX}\right)' \end{aligned}$$

For identity (55) let $Q = \{ \langle \tau \otimes \exp(-\sigma), \tau \otimes \exp(\sigma)^{-1} \rangle \mid \sigma, \tau \in \mathbb{R}^\omega \}$ and compute as follows:

$$\begin{aligned} (\tau \otimes \exp(-\sigma))' &= (\tau' - (\tau \otimes \sigma')) \otimes \exp(-\sigma) \\ Q &(\tau' - (\tau \otimes \sigma')) \otimes \exp(\sigma)^{-1} \\ &= (\tau' \otimes \exp(\sigma)^{-1}) - (\tau \otimes \sigma' \otimes \exp(\sigma)^{-1}) \\ &= (\tau' \otimes \exp(\sigma)^{-1}) - (\tau \otimes \sigma' \otimes \exp(\sigma) \otimes \exp(\sigma)^{-1} \otimes \exp(\sigma)^{-1}) \\ &= (\tau' \otimes \exp(\sigma)^{-1}) + (\tau \otimes (-\sigma' \otimes \exp(\sigma) \otimes \exp(\sigma)^{-1} \otimes \exp(\sigma)^{-1})) \\ &= (\tau' \otimes \exp(\sigma)^{-1}) + (\tau \otimes (\exp(\sigma)^{-1})') \\ &= (\tau \otimes \exp(\sigma)^{-1})' \end{aligned}$$

which proves that Q is a bisimulation. Identity (56) follows from identities (52) and (54). The remaining identities follow by induction on n . $\square$

9 Application: a divergent recurrence

As an addendum to Section 6, this very short section presents an example of a difference equation (recurrence relation) involving non-constant coefficients:

difference equation	initial value
$s_{n+1} - (n+1)s_n = 0$	$s_0 = 1$

One moment's thought suffices to see that the stream $(0!, 1!, 2!, \dots)$ is the solution of this equation. But as before, one would like to express this solution in terms of the constants and operators of stream calculus. As we shall see in a second, the two new operators of shuffle product and shuffle inverse will allow us to do so.

The stream $(0!, 1!, 2!, \dots)$ is the prototypical example of a divergent stream, in the sense that if one were to define a function $f(x) = 0! + 1!x + 2!x + \dots$, it would satisfy $f(0) = 0$, and would be undefined everywhere else. Therefore, the above recurrence cannot be solved with the use of generating functions (as in [Wil94]), which is the traditional approach in mathematics. The problem can be solved with the help of formal power series, but computing the solution of the present example leads to fairly complicated calculations, involving so-called hypergeometric series (cf. [GKP94, pp.346-348]).

In contrast, the solution is obtained in stream calculus in a surprisingly quick manner. Multiplying both sides of the behavioural differential equation with X^{n+1} (rather than X^n) and summing over all $n \geq 0$ gives

$$\sum_{n=0}^{\infty} s_{n+1} X^{n+1} - \sum_{n=0}^{\infty} (n+1) s_n X^{n+1} = 0$$

Putting as before $\sigma = (s_0, s_1, s_2, \dots)$ we find $\sigma - \sigma(0) = \sigma - 1$ for the first infinite sum and, using identity (49), $X \otimes \sigma$ for the second:

$$\sigma - 1 - X \otimes \sigma = 0$$

This gives $1 = \sigma - X \otimes \sigma = (1 - X) \otimes \sigma$. As a consequence,

$$\sigma = (1 - X)^{-1}$$

which by (51) is, indeed, what we expected: $\sigma = (0!, 1!, 2!, \dots)$.

10 Comparing convolution product and shuffle product

The relation between the two types of product and inverse is investigated. Its precise formulation will be in terms of a new type of stream derivation, which will later turn out to be useful for other purposes as well.

Let the *analytical* stream derivative of a stream σ be defined by

$$\frac{d}{dX}(\sigma) = (X \otimes \sigma)'$$

As usual, we shall also write

$$\frac{d\sigma}{dX} \equiv \frac{d}{dX}(\sigma)$$

There is the following general formula for our new type of derivation:

$$\frac{d}{dX}(s_0 + s_1 X^1 + s_2 X^2 + \dots) = s_1 + 2s_2 X^1 + 3s_3 X^2 + \dots$$

which we recognise as identity (50), and which explains the name of analytical stream derivation. Analytical stream derivation behaves for (convolution) product and inverse in the, from analysis, familiar way. In particular, there are the following identities:

$$\frac{d(\sigma \times \tau)}{dX} = \left(\frac{d\sigma}{dX} \times \tau\right) + \left(\sigma \times \frac{d\tau}{dX}\right)$$

$$\frac{d\sigma^{-1}}{dX} = -\frac{d\sigma}{dX} \times \sigma^{-1} \times \sigma^{-1}$$

$$\frac{d\sigma^{n+1}}{dX} = (n+1) \times \sigma' \times \sigma^n$$

Proof of identities (63)-(65): In the proof of identity (63), the following equalities are used (the proofs of these are left to the reader). For all summable families $\{\sigma_n\}_{n=0}^{\infty}$ and all τ in $\mathbb{R}^\omega$, for all $n \geq 0$,

$$\begin{aligned} \frac{d(\sum_{n=0}^{\infty} \sigma_n)}{dX} &= \sum_{n=0}^{\infty} \frac{d\sigma_n}{dX} \\ \frac{d(X^n \times \tau)}{dX} &= \left(\frac{dX^n}{dX} \times \tau\right) + (X^n \times \frac{d\tau}{dX}) \\ \frac{dX^{n+1}}{dX} &= (n+1) \times X^n \end{aligned}$$

Next consider $\sigma = (s_0, s_1, s_2, \dots)$ and τ in $\mathbb{R}^\omega$ and note that

$$\begin{aligned}
\frac{d(\sigma \times \tau)}{dX} &= \frac{d((\sum_{n=0}^{\infty} s_n \times X^n) \times \tau)}{dX} \quad [\text{Theorem 4.3}] \\
&= \frac{d(\sum_{n=0}^{\infty} s_n \times (X^n \times \tau))}{dX} \\
&= \sum_{n=0}^{\infty} \frac{d(s_n \times (X^n \times \tau))}{dX} \\
&= \sum_{n=0}^{\infty} s_n \times \frac{d(X^n \times \tau)}{dX} \\
&= \sum_{n=0}^{\infty} s_n \times \left(\frac{dX^n}{dX} \times \tau + X^n \times \frac{d\tau}{dX} \right) \\
&= \sum_{n=0}^{\infty} s_n \times \left(\frac{dX^n}{dX} \times \tau \right) + \sum_{n=0}^{\infty} s_n \times \left(X^n \times \frac{d\tau}{dX} \right) \\
&= \sum_{n=1}^{\infty} (n \times s_n \times X^{n-1} \times \tau) + \sum_{n=0}^{\infty} s_n \times \left(X^n \times \frac{d\tau}{dX} \right) \\
&= \left(\sum_{n=1}^{\infty} n \times s_n \times X^{n-1} \right) \times \tau + \left(\sum_{n=0}^{\infty} s_n \times X^n \right) \times \frac{d\tau}{dX} \\
&= \left(\frac{d\sigma}{dX} \times \tau \right) + \left(\sigma \times \frac{d\tau}{dX} \right) \quad [\text{identity (62) and Theorem 4.3}]
\end{aligned}$$

Identities (64) and (65) easily follow from (63). $\square$

Note that properties (63)-(65) reflect precisely how ordinary stream derivation behaves on shuffle product and shuffle inverse, which satisfy

$$\begin{aligned}
(\sigma \otimes \tau)' &= (\sigma' \otimes \tau) + (\sigma \otimes \tau') \\
(\sigma^{-1})' &= -\sigma' \otimes (\sigma^{-1} \otimes \sigma^{-1}) \\
(\sigma^{n+1})' &= (n+1) \otimes \sigma' \otimes \sigma^n
\end{aligned}$$

Analytical stream derivation is used in the following definition of an operator on streams:

behavioural differential equation	initial value
$\Lambda_c(\sigma)' = \Lambda_c(\frac{d\sigma}{dX})$	$\Lambda_c(\sigma)(0) = \sigma(0)$

One can easily prove that Λ_c transforms a stream $(s_0, s_1, s_2, s_3, \dots)$ into $(0!s_0, 1!s_1, 2!s_2, 3!s_3, \dots)$, or, equivalently,

$$\Lambda_c(s_0 + \frac{s_1}{1!}X + \frac{s_2}{2!}X^2 + \frac{s_3}{3!}X^3 + \dots) = s_0 + s_1X + s_2X^2 + s_3X^3 + \dots$$

In combinatorics, this is referred to as the *Laplace-Carson* transform (cf. [BLL98, p.350] and [Com74, p.48]), hence our notation. The following theorem shows that with Λ_c , we can relate the two types of product and inverse.

Theorem 10.1 *For all $r \in \mathbb{R}$, $\sigma, \tau \in \mathbb{R}^\omega$,*

$$\begin{aligned}
\Lambda_c(\frac{d\sigma}{dX}) &= \Lambda_c(\sigma)' \\
\Lambda_c(r) &= r \\
\Lambda_c(X) &= X \\
\Lambda_c(\sigma + \tau) &= \Lambda_c(\sigma) + \Lambda_c(\tau) \\
\Lambda_c(\sigma \times \tau) &= \Lambda_c(\sigma) \otimes \Lambda_c(\tau) \\
\Lambda_c(\sigma^{-1}) &= \Lambda_c(\sigma)^{-1}
\end{aligned}$$

Proof: The first equality is by definition and the next two are trivial. For the latter three, let $R \subseteq \mathbb{R}^\omega \times \mathbb{R}^\omega$ be the smallest relation such that

1. $\{\langle \sigma, \sigma \rangle \mid \sigma \in \mathbb{R}^\omega\} \subseteq R$
2. for all $\sigma_1, \sigma_2, \tau_1, \tau_2 \in \mathbb{R}^\omega$, if $\langle \Lambda_c(\sigma_1), \tau_1 \rangle \in R$ and $\langle \Lambda_c(\sigma_2), \tau_2 \rangle \in R$ then
 - (a) $\langle \Lambda_c(\sigma_1 + \sigma_2), \tau_1 + \tau_2 \rangle \in R$
 - (b) $\langle \Lambda_c(\sigma_1 \times \sigma_2), \tau_1 \otimes \tau_2 \rangle \in R$
 - (c) $\langle \Lambda_c((\sigma_1)^{-1}), (\tau_1)^{-1} \rangle \in R$

Then R is a bisimulation on $\mathbb{R}^\omega$ and the result follows by coinduction. $\square$

Thus the operator Λ_c allows one to switch between the two different ring structures on $\mathbb{R}^\omega$ that are determined by convolution product and shuffle product, each of which comes along with its own type of (partially defined) operation of inverse, and its own type of derivative:

$$\Lambda_c : \langle \mathbb{R}^\omega, +, \times, (-)^{-1}, \frac{d(-)}{dX} \rangle \rightarrow \langle \mathbb{R}^\omega, +, \otimes, (-)^{-1}, (-)' \rangle$$

Let us emphasize, however, that for the stream calculus we are developing, it is of crucial importance to have both structures present at the same time. Notably the interplay between the various operators from both worlds, turns out to constitute the most interesting part of the calculus. The following identities clearly illustrate this point, since they involve both the shuffle product and the (convolution) inverse. They all have in common that they provide a way of eliminating the occurrence of the shuffle product, a procedure we shall sometimes refer to as *shuffle elimination*.

Theorem 10.2 *For all $r \in \mathbb{R}$ and $\sigma \in \mathbb{R}^\omega$:*

$$1 = \frac{1}{1 + rX} \otimes \frac{1}{1 - rX} \quad (66)$$

$$X \otimes \sigma = (X^2 \times \frac{d\sigma}{dX}) + (X \times \sigma) \quad (67)$$

$$(\frac{1}{1 - rX}) \otimes \sigma = (\frac{1}{1 - rX}) \times (\sigma \circ \frac{X}{1 - rX}) \quad (68)$$

$$= \frac{s_0}{1 - rX} + \frac{s_1 X}{(1 - rX)^2} + \frac{s_2 X^2}{(1 - rX)^3} + \dots \quad (69)$$

Proof: Identity (66) follows from (56). For (67), observe that

$$\begin{aligned} & (X^2 \times \frac{d\sigma}{dX}) + (X \times \sigma) \\ &= (X^2 \times (X \otimes \sigma'))' + (X \times \sigma) \\ &= (X \times (X \times (X \otimes \sigma'))') + (X \times \sigma) \\ &= (X \times (X \otimes \sigma')) + (X \times \sigma) \quad [\text{Fundamental Theorem 4.1, } (X \otimes \sigma')(0) = 0] \\ &= X \times ((X \otimes \sigma') + \sigma) \\ &= X \times ((X \otimes \sigma)') \\ &= X \otimes \sigma \quad [\text{Fundamental Theorem 4.1, } (X \otimes \sigma)(0) = 0] \end{aligned}$$

And for (68), define

$$R = \{ \langle (\frac{s}{1 - rX}) \otimes \sigma, (\frac{s}{1 - rX}) \times (\sigma \circ \frac{X}{1 - rX}) \rangle \mid r, s \in \mathbb{R}, \sigma \in \mathbb{R}^\omega \}$$

and compute as follows to see that R is a bisimulation-up-to:

$$\begin{aligned}
((\frac{s}{1-rX}) \otimes \sigma)' &= (\frac{r \times s}{1-rX}) \otimes \sigma + (\frac{s}{1-rX}) \otimes \sigma' \\
\Sigma R \quad (\frac{r \times s}{1-rX}) \times (\sigma \circ \frac{X}{1-rX}) &+ (\frac{s}{1-rX}) \times (\sigma' \circ \frac{X}{1-rX}) \\
&= (\frac{s}{1-rX})' \times (\sigma \circ \frac{X}{1-rX}) + (s \times (\sigma \circ \frac{X}{1-rX}))' \\
&= ((\frac{s}{1-rX}) \times (\sigma \circ \frac{X}{1-rX}))'
\end{aligned}$$

Identity (68) then follows by coinduction-up-to. Identity (69) follows from (68) and (25). $\square$

To illustrate identity (67), we compute

$$\begin{aligned}
X \otimes \frac{1}{1+X^2} &= X^2 \times \frac{d}{dX}(\frac{1}{1+X^2}) + (X \times \frac{1}{1+X^2}) \\
&= X^2 \times \frac{-2X}{(1+X^2)^2} + \frac{X}{1+X^2} \quad [\text{identity (63)}] \\
&= \frac{X - X^3}{(1+X^2)^2}
\end{aligned}$$

Similar computations give

$$\begin{aligned}
X \otimes \frac{X}{1+X^2} &= \frac{2X^2}{(1+X^2)^2} \\
X^2 \otimes \frac{1}{1+X^2} &= \frac{X^2 - 3X^4}{(1+X^2)^3} \\
X^2 \otimes \frac{X}{1+X^2} &= \frac{-3X^3 + X^5}{(1+X^2)^3}
\end{aligned}$$

The following two identities are useful special cases of (68):

$$X^n \otimes \frac{1}{1-rX} = \frac{X^n}{(1-rX)^{n+1}} \quad (70)$$

$$(1+rX)^n \otimes \frac{1}{1-rX} = \frac{1}{(1-rX)^{n+1}} \quad (71)$$

11 Application: a generalised Euler formula

As an illustration of the use of stream exponentiation and shuffle elimination, we present a quick stream calculus derivation of a so-called generalised Euler formula.

Let the difference $\Delta\sigma$ of a stream $\sigma = (s_0, s_1, s_2, \dots)$ be defined as

$$\begin{aligned}
\Delta\sigma &= \sigma' - \sigma \\
&= (s_1 - s_0, s_2 - s_1, s_3 - s_2, \dots)
\end{aligned} \quad (72)$$

Defining, as usual, $\Delta^0\sigma = \sigma$ and $\Delta^{n+1}\sigma = \Delta(\Delta^n\sigma)$, for all $n \geq 0$, we have the following identity:

$$\sigma \otimes \frac{1}{1+X} = (\Delta^0\sigma)(0) + (\Delta^1\sigma)(0) \times X^1 + (\Delta^2\sigma)(0) \times X^2 + \dots \quad (73)$$

Proof of (73): First note that for all $n \geq 0$,

$$\begin{aligned}
\Delta^{n+1}\sigma &= \Delta(\Delta^n\sigma) \\
&= (\Delta^n\sigma)' - \Delta^n\sigma \\
&= \Delta^n(\sigma') - \Delta^n\sigma
\end{aligned}$$

Next define $R = \{ \langle \sigma \otimes \frac{1}{1+X}, (\Delta^0 \sigma)(0) + (\Delta^1 \sigma)(0) \times X^1 + (\Delta^2 \sigma)(0) \times X^2 + \dots \rangle \mid \sigma \in \mathbb{R}^\omega \}$ and compute as follows to see that R is a bisimulation-up-to:

$$\begin{aligned}
(\sigma \otimes \frac{1}{1+X})' &= (\sigma' \otimes \frac{1}{1+X}) - (\sigma \otimes \frac{1}{1+X}) \\
\Sigma R & ((\Delta^0 \sigma')(0) + (\Delta^1 \sigma')(0) \times X^1 + (\Delta^2 \sigma')(0) \times X^2 + \dots) - \\
& ((\Delta^0 \sigma)(0) + (\Delta^1 \sigma)(0) \times X^1 + (\Delta^2 \sigma)(0) \times X^2 + \dots) \\
&= (\Delta^0 \sigma' - \Delta^0 \sigma)(0) + \\
& (\Delta^1 \sigma' - \Delta^1 \sigma)(0) \times X^1 + \\
& (\Delta^2 \sigma' - \Delta^2 \sigma)(0) \times X^2 + \dots \\
&= (\Delta^1 \sigma)(0) + (\Delta^2 \sigma)(0) \times X^1 + (\Delta^3 \sigma)(0) \times X^2 + \dots \\
&= ((\Delta^0 \sigma)(0) + (\Delta^1 \sigma)(0) \times X^1 + (\Delta^2 \sigma)(0) \times X^2 + \dots)'
\end{aligned}$$

Identity (73) now follows by coinduction-up-to. $\square$

Equivalently, we have

$$\sigma \otimes \frac{1}{1+X} = s_0 + (s_1 - s_0)X + (s_2 - 2s_1 + s_0)X^2 + (s_3 - 3s_2 + 3s_1 - s_0)X^3 + \dots$$

which is immediate from the fact that, for all $n \geq 0$,

$$\Delta^n \sigma = \sum_{k=0}^n (-1)^k \binom{n}{k} \sigma^{(n-k)}$$

Using identity (73), the following theorem can now be proved.

Theorem 11.1 *For any stream σ ,*

$$\sigma = \frac{(\Delta^0 \sigma)(0)}{1-X} + \frac{(\Delta^1 \sigma)(0) \times X^1}{(1-X)^2} + \frac{(\Delta^2 \sigma)(0) \times X^2}{(1-X)^3} + \dots$$

Proof:

$$\begin{aligned}
\sigma &= \sigma \otimes 1 \\
&= \sigma \otimes \frac{1}{1+X} \otimes \frac{1}{1-X} \quad [\text{identity (66)}] \\
&= ((\Delta^0 \sigma)(0) + (\Delta^1 \sigma)(0) \times X^1 + (\Delta^2 \sigma)(0) \times X^2 + \dots) \otimes \frac{1}{1-X} \quad [\text{identity (73)}] \\
&= \frac{(\Delta^0 \sigma)(0)}{1-X} + \frac{(\Delta^1 \sigma)(0) \times X^1}{(1-X)^2} + \frac{(\Delta^2 \sigma)(0) \times X^2}{(1-X)^3} + \dots \quad [\text{identity (69), for } r=1]
\end{aligned}$$

$\square$

The reader is invited to compare this derivation with the one of, for instance, [Sch68, 11.38], where the same formula can be derived only under a number of convergence assumptions.

12 Application: solving analytical differential equations

A classical technique in analysis for the solution of differential equations defining analytical functions, is the method of undetermined coefficients (cf. [BR78, p.82]). The idea is quickly explained by means of an example. In order to solve the differential equation

differential equation	initial value
$f'' + f = 0$	$f(0) = 0, f'(0) = 1$

one assumes the solution to be of the shape

$$f(x) = s_0 + \frac{s_1}{1!}x + \frac{s_2}{2!}x^2 + \dots$$

Computing f'' gives

$$f''(x) = s_2 + \frac{s_3}{1!}x + \frac{s_4}{2!}x^2 + \dots$$

Substituting the expressions for f and f'' in the differential equation, one obtains the following difference equation for the coefficients $(s_0, s_1, s_2, \dots)$ of f :

difference equation	initial value
$s_{n+2} + s_n = 0$	$s_0 = 0, s_1 = 1$

Thus this method reduces the problem of solving a differential equation for f to the problem of solving a difference equation for the Taylor coefficients of f . Though conceptually very simple, the method of undetermined coefficients has two major drawbacks. Firstly, more interesting differential equations quickly lead to very complicated difference equations. Secondly, there is no general technique for translating the solution of the difference equation (if found at all) back into a workable expression for f .

Here we shall present a variant of the method above, which in many applications is free of these restrictions. Defining

$$\mathcal{A} = \{f : \mathbb{R} \rightarrow \mathbb{R} \mid f \text{ is analytic in (a neighbourhood of) } 0 \}$$

our main tool will be the function $\mathcal{T} : \mathcal{A} \rightarrow \mathbb{R}^\omega$, which sends an analytic function f to its Taylor series $\mathcal{T}(f) = (f(0), f'(0), f''(0), \dots)$. Formally, $\mathcal{T}(f)$ is defined by the following system of behavioural differential equations (one for each $f \in \mathcal{A}$):

behavioural differential equation	initial value
$\mathcal{T}(f)' = \mathcal{T}(f')$	$\mathcal{T}(f)(0) = f(0)$

(Note that the first occurrence of $'$ in $\mathcal{T}(f)' = \mathcal{T}(f')$ stands for stream derivation, whereas the second denotes analytical function derivation.) This definition is a variation on a definition by Pavlović and Escardó [PE98], who characterised $\mathcal{T}$ as a final coalgebra homomorphism, in order to give a coinductive characterization of the Laplace transform. Here $\mathcal{T}$ is studied in its own right, and will serve rather as an alternative to Laplace transform. For that reason, we shall sometimes refer to the stream $\mathcal{T}(f)$ as the *Taylor transform* of f .

Our method is characterised by the following three steps:

- (1) The function $\mathcal{T}$ is used to transform, in a systematic fashion, a differential equation for f into a behavioural differential equation for the stream $\mathcal{T}(f)$ of Taylor coefficients of f .
- (2) The behavioural differential equation is solved in stream calculus by means of the techniques of Section 5.
- (3) The resulting solution is translated back in a systematic manner into an expression for f .

First of all, the following theorem expresses that the function $\mathcal{T}$ transforms functions into their Taylor series in a systematic manner, indeed. For functions f and g , we shall be using the following familiar definitions: for all $x \in \mathbb{R}$,

$$(f + g)(x) = f(x) + g(x), \quad (f \cdot g)(x) = f(x) \times g(x), \quad f^{-1}(x) = f(x)^{-1}, \quad e^f(x) = e^{f(x)}$$

Theorem 12.1 *For all analytic functions $f, g \in \mathcal{A}$,*

$$\mathcal{T}(f)(0) = f(0) \tag{74}$$

$$\mathcal{T}(f') = \mathcal{T}(f)' \tag{75}$$

$$\mathcal{T}(f + g) = \mathcal{T}(f) + \mathcal{T}(g) \tag{76}$$

$$\mathcal{T}(f \cdot g) = \mathcal{T}(f) \otimes \mathcal{T}(g) \tag{77}$$

$$\mathcal{T}(f^{-1}) = \mathcal{T}(f)^{-1} \tag{78}$$

$$\mathcal{T}(e^f) = \exp(\mathcal{T}(f)) \tag{79}$$

The most useful instantiation of identity (79) will be the case that $f(x) = rx$: see identity (82) below. Identities (74)-(77) are implicitly present in [PE98]; see also [Rut00a].

Proof: Identities (74) and (75) are immediate by the definition of $\mathcal{T}$. For the others we use coinduction: let $R \subseteq \mathbb{R}^\omega \times \mathbb{R}^\omega$ be the smallest relation on streams such that $\langle \mathcal{T}(f), \mathcal{T}(f) \rangle \in R$, for all $f \in \mathcal{A}$, and such that if $\langle \mathcal{T}(f_1), \mathcal{T}(f_2) \rangle \in R$ and $\langle \mathcal{T}(g_1), \mathcal{T}(g_2) \rangle \in R$ then

1. $\langle \mathcal{T}(f_1 + g_1), \mathcal{T}(f_2) + \mathcal{T}(g_2) \rangle \in R$
2. $\langle \mathcal{T}(f_1 \cdot g_1), \mathcal{T}(f_2) \otimes \mathcal{T}(g_2) \rangle \in R$
3. $\langle \mathcal{T}(f_1^{-1}), \mathcal{T}(f_2)^{-1} \rangle \in R$
4. $\langle \mathcal{T}(e^{f_1}), \exp(\mathcal{T}(f_2)) \rangle \in R$

The relation R is a bisimulation: consider for instance $\langle \mathcal{T}(f \cdot g), \mathcal{T}(f) \otimes \mathcal{T}(g) \rangle \in R$. Both streams clearly have the same initial value. And $\langle \mathcal{T}(f \cdot g)', (\mathcal{T}(f) \otimes \mathcal{T}(g))' \rangle \in R$, since

$$\begin{aligned}
& \mathcal{T}(f \cdot g)' \\
&= \mathcal{T}((f \cdot g)') \\
&= \mathcal{T}(f' \cdot g + f \cdot g') \\
&\stackrel{R}{=} \mathcal{T}(f') \otimes \mathcal{T}(g) + \mathcal{T}(f) \otimes \mathcal{T}(g') \\
&= \mathcal{T}(f)' \otimes \mathcal{T}(g) + \mathcal{T}(f) \otimes \mathcal{T}(g)' \\
&= (\mathcal{T}(f) \otimes \mathcal{T}(g))'
\end{aligned}$$

Similarly for the other elements of R . Now the theorem follows by coinduction. $\square$

The following set of identities on the Taylor transforms of some well known functions will be useful when solving differential equations.

Theorem 12.2 *For all $r \in \mathbb{R}$, $n \geq 0$, $f \in \mathcal{A}$,*

$$\mathcal{T}(r) = r \tag{80}$$

$$\mathcal{T}(x^n) = n! X^n \tag{81}$$

$$\mathcal{T}(e^{rx}) = \frac{1}{1 - rX} \tag{82}$$

$$\mathcal{T}(\sin(rx)) = \frac{rX}{1 + r^2 X^2} \tag{83}$$

$$\mathcal{T}(\cos(rx)) = \frac{1}{1 + r^2 X^2} \tag{84}$$

$$\mathcal{T}(f \times e^{rx}) = \left(\frac{1}{1 - rX} \right) \times (\mathcal{T}(f) \circ \frac{X}{1 - rX}) \tag{85}$$

$$\mathcal{T}(f \times x) = \left(X^2 \times \frac{d\mathcal{T}(f)}{dX} \right) + (X \times \mathcal{T}(f)) \tag{86}$$

Proof: The first two equalities are straightforward. For (82)-(84), define

$$\begin{aligned}
R = & \{ \langle \mathcal{T}(s \times e^{rx}), \frac{s}{1 - rX} \rangle \mid r, s \in \mathbb{R} \} \cup \\
& \{ \langle \mathcal{T}(s \times \sin(rx)), \frac{(s \times r)X}{1 + r^2 X^2} \rangle \mid r, s \in \mathbb{R} \} \cup \\
& \{ \langle \mathcal{T}(s \times \cos(rx)), \frac{s}{1 + r^2 X^2} \rangle \mid r, s \in \mathbb{R} \}
\end{aligned}$$

and use coinduction. For (85), note that

$$\begin{aligned}
\mathcal{T}(f \times e^{rx}) &= \mathcal{T}(f) \otimes \mathcal{T}(e^{rx}) \quad [\text{identity (77)}] \\
&= \mathcal{T}(f) \otimes \left(\frac{1}{1-rX}\right) \quad [\text{identity (82)}] \\
&= \left(\frac{1}{1-rX}\right) \times (\mathcal{T}(f) \circ \frac{X}{1-rX}) \quad [\text{identity (68)}]
\end{aligned}$$

For (86), finally, we have:

$$\begin{aligned}
\mathcal{T}(f \times x) &= \mathcal{T}(f) \otimes \mathcal{T}(x) \quad [\text{identity (77)}] \\
&= \mathcal{T}(f) \otimes X \quad [\text{identity (81)}] \\
&= (X^2 \times \frac{d\mathcal{T}(f)}{dX}) + (X \times \mathcal{T}(f)) \quad [\text{identity (67)}]
\end{aligned}$$

□

By now, we are sufficiently prepared to tackle a variety of differential equations:

1. As a warming up, consider the equation from the beginning of the present section:

differential equation	initial value
$f'' + f = 0$	$f(0) = 0, f'(0) = 1$

Applying $\mathcal{T}$ to both sides of the equation, and putting $\sigma = \mathcal{T}(f)$ gives

behavioural differential equation	initial value
$\sigma'' + \sigma = 0$	$\sigma(0) = 0, \sigma'(0) = 1$

This constitutes step (1) of our method. In step (2), the resulting behavioural differential equation is solved according to the techniques of Section 5, yielding

$$\sigma = \frac{X}{1+X^2}$$

In order to translate this outcome back into the function f —step (3) of our method—we can apply identity (24), yielding

$$\sigma = X - X^3 + X^5 - X^7 + \dots$$

which, in combination with identity (81) gives

$$f(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$$

Often, we can improve on the last step. In this particular example, it is sufficient to consult our set of basic identities on Taylor transforms, to find that identity (83) provides us immediately with an answer:

$$f(x) = \sin(x)$$

2. For a second example, consider the following non-homogeneous equation:

differential equation	initial value
$f' - f = e^x$	$f(0) = 1$

Using identity (82) and writing $\sigma = \mathcal{T}(f)$, step (1) transforms this equation into

behavioural differential equation	initial value
$\sigma' - \sigma = (1 - X)^{-1}$	$\sigma(0) = 1$

The solution of this equation, step (2), is identity (27). For step (3), we combine $\sigma = \mathcal{T}(f)$ with identities (22) and (81), yielding

$$f(x) = 1 + \frac{2}{1!}x + \frac{3}{2!}x^2 + \frac{4}{3!}x^3 \dots$$

Again, we can do better than this. Using some elementary stream calculus, we can rewrite σ as follows:

$$\begin{aligned} \sigma &= \frac{1}{(1 - X)^2} \\ &= (1 + X) \otimes \frac{1}{1 - X} \quad [\text{by identity (71)}] \end{aligned}$$

Using now identities (76), (77), and (82), we get

$$f(x) = (1 + x) \cdot e^x$$

3. Here is example number three:

differential equation	initial value
$f'' + r^2 f = 1$	$f(0) = 1, f'(0) = 1$

Putting $\sigma = \mathcal{T}(f)$, step (1) gives

behavioural differential equation	initial value
$\sigma'' + r^2 \sigma = 1$	$\sigma(0) = 1, \sigma'(0) = 1$

For step (2) we recall the solution of this equation from Section 5, and perform some elementary stream calculus on it:

$$\begin{aligned} \sigma &= \frac{1 + X + X^2}{1 + r^2 X^2} \\ &= \frac{1}{1 + r^2 X^2} + \frac{X}{1 + r^2 X^2} + \frac{X^2}{1 + r^2 X^2} \\ &= \frac{1}{1 + r^2 X^2} + \frac{1}{r} \frac{rX}{1 + r^2 X^2} + \frac{1}{r^2} \left(1 - \frac{1}{1 + r^2 X^2}\right) \end{aligned}$$

The rewriting was done to make step (3) easy: applying identities (83) and (84) yields the final outcome:

$$f(x) = \cos(rx) + \frac{1}{r} \sin(rx) + \frac{1}{r^2} (1 - \cos(rx))$$

4. Example number four:

differential equation	initial value
$f'' - f' = 2 + 6x$	$f(0) = 1, f'(0) = 0$

Step (1) gives

behavioural differential equation	initial value
$\sigma'' - \sigma' = 2 + 6X$	$\sigma(0) = 1, \sigma'(0) = 0$

As before, we recall the solution of this equation (step (2)) from Section 5, and perform some elementary stream calculus on it:

$$\begin{aligned}\sigma &= \frac{1 - X + 2X + 6X^3}{1 - X} \\ &= -7 - 8X - 6X^2 + 8\frac{1}{1 - X} \quad [\text{applying identity (60) four times}]\end{aligned}$$

Applying the by now familiar identities on the Taylor transforms, step (3) yields

$$f(x) = -7 - 8x - 3x^2 + 8e^x$$

13 Weighted stream automata

In this section, the notion of weighted stream automaton is introduced (called nondeterministic automaton in [Rut00a]), the transition diagrams of which can serve as pleasant graphical representations of streams and their successive derivatives. Finite such automata turn out to correspond to rational streams, but as we shall see, infinite weighted automata will be useful too.

A $\mathbb{R}$ -*weighted stream automaton*, or *weighted automaton* for short, is a pair $Q = (Q, \langle o, t \rangle)$ consisting of a set Q of states, together with an output function $o : Q \rightarrow \mathbb{R}$, and a transition function $t : Q \rightarrow (Q \rightarrow_f \mathbb{R})$, where the latter set contains only functions of finite *support*:

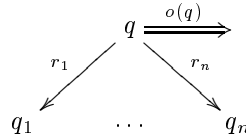
$$Q \rightarrow_f \mathbb{R} = \{\phi : Q \rightarrow \mathbb{R} \mid |\{q \in Q \mid \phi(q) \neq 0\}| < \infty\}$$

The output function o assigns to each state q in Q a real number $o(q)$ in $\mathbb{R}$. The transition function t assigns to a state q in Q a function $t(q) : Q \rightarrow_f \mathbb{R}$, which specifies for any state q' in Q a real number $t(q)(q')$ in $\mathbb{R}$. This number can be thought of as the weight (cost, multiplicity, duration, etc.) with which the transition from q to q' occurs. The following notation will be used:

$$q \xrightarrow{x} q' \text{ iff } t(q)(q') = x$$

$$q \xRightarrow{x} \text{ iff } o(q) = x$$

which will allow us to present weighted automata by pictures. In such pictures, only those arrows will be drawn that have a non-zero label. For $q \in Q$, let $\{q_1, \dots, q_n\}$ be the support of $t(q)$, that is, the set of all states q_i for which $t(q)(q_i) \neq 0$, and let $r_i = t(q)(q_i)$, for $1 \leq i \leq n$. The following diagram contains all the relevant information about the state q in the automaton Q :



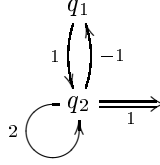
Note that the requirement of finite support implies that the automaton Q is *finitely branching*, in the sense that from q , there are only finitely many (non-zero) arrows.

The *stream behaviour* of a state q in a weighted automaton Q is the stream $S(q) \in \mathbb{R}^\omega$, defined coinductively by the following system of behavioural differential equations (one for each state in Q):

differential equation	initial value
$S(q)' = r_1 S(q_1) + \dots + r_n S(q_n)$	$S(q)(0) = o(q)$

(As before, $\{q_1, \dots, q_n\}$ is the support of $t(q)$ and $r_i = t(q)(q_i)$, for $1 \leq i \leq n$.) The pair (Q, q) is called a *representation* of the stream $S(q)$. A stream $\sigma \in \mathbb{R}^\omega$ is called *finitely representable* if there exists a *finite* weighted automaton Q and $q \in Q$ with $\sigma = S(q)$. Such streams are also called *recognizable*.

For a simple example, consider the following two state weighted automaton:



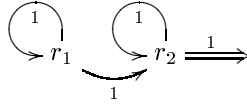
We have the following equations for the behaviour of q_1 and q_2 :

differential equation	initial value
$S(q_1)' = S(q_2)$	$S(q_1)(0) = 0$
$S(q_2)' = -S(q_1) + 2S(q_2)$	$S(q_2)(0) = 1$

Applying the methodology of Section 5 (generalised to *systems* of equations), we find:

$$S(q_1) = \frac{X}{(1-X)^2}, \quad S(q_2) = \frac{1}{(1-X)^2}$$

Thus the stream $X(1-X)^{-2}$ is represented by the state q_1 above. Note that representations are by no means unique: $X(1-X)^{-2}$ is also represented by the state r_1 in the following automaton:



since one readily computes that $S(r_1) = X(1-X)^{-2}$ and $S(r_2) = (1-X)^{-1}$.

The following proposition describes how the behaviour of a state of a weighted automaton can be computed in terms of the labels of its transition sequences. Although this proposition provides some operational intuition about the behaviour of weighted automata, it is not very suited for reasoning about them; it will play no role in the remainder of this paper.

Proposition 13.1 *For a weighted automaton Q , for all $q \in Q$ and $k \geq 0$,*

$$S(q)(k) = \sum \{l_0 l_1 \cdots l_{k-1} l \mid q = q_0 \xrightarrow{l_0} q_1 \xrightarrow{l_1} \cdots \xrightarrow{l_{k-1}} q_k \xrightarrow{l} \}$$

For instance, in the last example of the (two state) automaton above, one readily verifies that $S(r_1)(n) = n$, for all $n \geq 0$, which is consistent with what we found earlier, since $(0, 1, 2, 3, \dots) = X(1-X)^{-2}$.

Proof: Using the differential equation for $S(q)$ and the observation that

$$\begin{aligned}
S(q)(k+1) &= S(q)^{(k+1)}(0) \\
&= (S(q)')^{(k)}(0) \\
&= (r_1 S(q_1) + \cdots + r_n S(q_n))^{(k)}(0) \\
&= r_1 S(q_1)^{(k)}(0) + \cdots + r_n S(q_n)^{(k)}(0) \\
&= r_1 S(q_1)(k) + \cdots + r_n S(q_n)(k)
\end{aligned}$$

the proof follows by induction on k . □

There is also the following algebraic characterisation of the behaviour of a *finite* weighted automaton $Q = (Q, \langle o, t \rangle)$. It will play no role in the remainder of this paper. Let $Q = \{q_1, \dots, q_n\}$ and let μ be the $n \times n$ matrix with entries $\mu_{ij} = t(q_i)(q_j)$. Furthermore write $o : Q \rightarrow \mathbb{R}$ as a column vector $o^t = (o(q_1), \dots, o(q_n))^t$.

Proposition 13.2 *For any sequence of real numbers $a = (a_1, \dots, a_n)$ (viewed as a row vector), and for all $k \geq 0$,*

$$a_1 S(q_1)(k) + \dots + a_n S(q_n)(k) = a \times \mu^k \times o^t$$

where on the right, matrix multiplication is used.

Proof: The proof is an immediate consequence of Proposition 13.1. □

The streams that can be represented by finite weighted automata are precisely the *rational* streams, which are introduced and characterised next.

Theorem 13.3 *The following three conditions are equivalent. Any stream σ satisfying them is called rational.*

- (1) *There exist polynomial streams π and ρ with $\rho(0) \neq 0$ such that*

$$\sigma = \frac{\pi}{\rho}$$

- (2) *There exists $n \geq 1$, real numbers s_{ij} , for $1 \leq i \leq n$ and $1 \leq j \leq n$, real numbers $k_1, \dots, k_n$, and streams $\sigma_1, \sigma_2, \dots, \sigma_n$ with $\sigma = \sigma_1$, satisfying the following finite system of behavioural differential equations:*

behavioural differential equation	initial value
$(\sigma_1)' = s_{11}\sigma_1 + s_{12}\sigma_2 + \dots + s_{1n}\sigma_n$	$\sigma_1(0) = k_1$
$(\sigma_2)' = s_{21}\sigma_1 + s_{22}\sigma_2 + \dots + s_{2n}\sigma_n$	$\sigma_2(0) = k_2$
$\dots$	$\dots$
$(\sigma_n)' = s_{n1}\sigma_1 + s_{n2}\sigma_2 + \dots + s_{nn}\sigma_n$	$\sigma_n(0) = k_n$

- (3) *σ has a finite representation (is recognizable): There exist a finite weighted automaton Q and a state $q \in Q$ with $\sigma = S(q)$.*

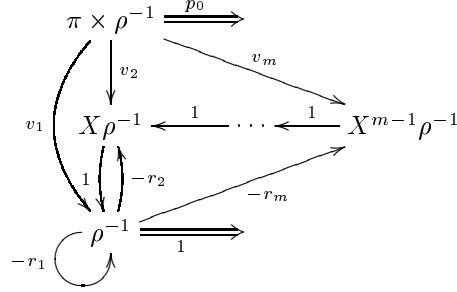
Proof: A proof can be found in [Rut00a]. (See also Section 18 for further references on rationality.) Most interesting is the proof that (1) implies (2), which is as follows. Consider polynomials $\pi = p_0 + p_1X + \dots + p_nX^n$ and $\rho = r_0 + r_1X + \dots + r_mX^m$ and let $\sigma = \pi \times \rho^{-1}$. For notational convenience, we assume that $r_0 = 1$. We also assume that $0 < n < m$ (the case that $m \leq n$ can be dealt with similarly). Using the defining behavioural differential equations of sum, convolution product, and inverse, σ is readily seen to satisfy the following finite system of behavioural differential equations:

behavioural differential equation	initial value
$(\pi \times \rho^{-1})' = v_1\rho^{-1} + v_2X\rho^{-1} + \dots + v_mX^{m-1}\rho^{-1}$	$(\pi \times \rho^{-1})(0) = p_0$
$(\rho^{-1})' = -r_1\rho^{-1} - r_2X\rho^{-1} - \dots - r_mX^{m-1}\rho^{-1}$	$(\rho^{-1})(0) = 1$
$(X\rho^{-1})' = \rho^{-1}$	$(X\rho^{-1})(0) = 0$
$\dots$	$\dots$
$(X^{m-1}\rho^{-1})' = X^{m-2}\rho^{-1}$	$(X^{m-1}\rho^{-1})(0) = 0$

where

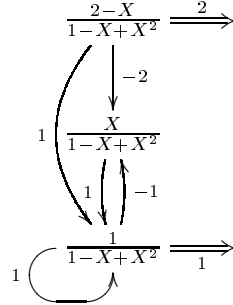
$$v_i = \begin{cases} p_i - p_0 r_i & \text{if } 1 \leq i \leq n \\ -p_0 r_i & \text{if } n < i \leq m \end{cases}$$

The equivalence between (2) and (3) follows from the correspondence between finite systems of behavioural differential equations and finite representations. More concretely, the above system of equations for $\pi \times \rho^{-1}$ corresponds to the following weighted automaton:



□

Note that in the above construction of a weighted automaton for rational streams, we have used the streams themselves ($\pi \times \rho^{-1}$, $X\rho^{-1}$, etc.) as the states of this automaton. The transitions of any stream (state) in this automaton are obtained by ‘splitting the derivative’ of this stream into its ‘+’ components. Here is an example, for $\pi = 2 - X$ and $\rho = 1 - X + X^2$:



In this automaton, the transitions of for instance the lower state are determined by the fact that its derivative is a sum of two streams, as follows:

$$\left(\frac{1}{1-X+X^2}\right)' = -\frac{X}{1-X+X^2} + \frac{1}{1-X+X^2}$$

Later we shall apply this procedure of ‘splitting derivatives’ also in the construction of infinite weighted automata for non-rational streams.

Recall that in Theorem 10.2, a number of identities for shuffle elimination was proved. The following theorem expresses that, as a consequence of Theorem 13.3 above, we now know how to eliminate shuffle products of any two rational streams.

Theorem 13.4 *If σ and τ are rational streams then $\sigma \otimes \tau$ is rational too.*

Proof: Consider two rational streams σ and τ . According to Theorem 13.3, there exist finite systems of behavioural differential equations

$$\{(\sigma_i)' = s_{i1}\sigma_1 + \dots + s_{in}\sigma_n \mid \sigma_i(0) = k_i\}_{i=1}^n$$

and

$$\{(\tau_i)' = t_{i1}\tau_1 + \cdots + t_{im}\tau_m \mid \tau_i(0) = l_i\}_{i=1}^m$$

which are satisfied by $\sigma = \sigma_1$ and $\tau = \tau_1$. It is an immediate consequence of the definition of the shuffle product that $\sigma \otimes \tau = \sigma_1 \otimes \tau_1$ satisfies the following finite system of behavioural differential equations:

$$\{(\sigma_i \otimes \tau_j)' = s_{i1}(\sigma_1 \otimes \tau_j) + \cdots + s_{in}(\sigma_n \otimes \tau_j) + t_{j1}(\sigma_i \otimes \tau_1) + \cdots + t_{jm}(\sigma_i \otimes \tau_m)\}_{i,j}$$

with initial values $\{(\sigma_i \otimes \tau_j)(0) = k_i \times l_j\}_{i,j}$. Thus $\sigma \otimes \tau$ is rational, again by Theorem 13.3. $\square$

The proof of Theorem 13.4 in fact describes a general algorithm for the shuffle elimination for rational streams, which is illustrated by the following example. Consider two streams $\sigma = (1 + X^2)^{-1}$ and $\tau = X(1 + X^2)^{-1}$. Since $\sigma' = -\tau$ and $\tau' = \sigma$, we have the following system of equations for $\sigma \otimes \tau$:

$$\begin{aligned} (\sigma \otimes \tau)' &= -\tau \otimes \tau + \sigma \otimes \sigma \\ (\sigma \otimes \sigma)' &= -2\sigma \otimes \tau \\ (\tau \otimes \tau)' &= 2\sigma \otimes \tau \end{aligned}$$

with the obvious initial values. Solving this system of behavioural differential equations, one finds

$$\frac{1}{1+X^2} \otimes \frac{X}{1+X^2} = \frac{X}{1+4X^2}$$

Note that rational streams are *not* closed under shuffle inverse. A basic example is the stream

$$(1 - X)^{-1} = (0!, 1!, 2!, \dots)$$

which can be shown to be non-rational. (A proof would be based on the fact that the elements of any rational stream can be expressed in terms of some polynomial expression, that can never ‘grow’ as fast as the stream of factorials.)

One of the advantages of weighted automata is that they provide finite representations for rational streams. It turns out to be worthwhile to study also infinite weighted automata representing non-rational streams. We hope the next example convinces the reader hereof.

The function $\tan(x)$ satisfies the following (ordinary) differential equation:

differential equation	initial value
$\tan' = 1 + \tan^2$	$\tan(0) = 0$

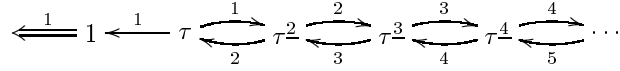
Putting $\tau = \mathcal{T}(\tan(x))$ for the Taylor series of $\tan(x)$, and recalling identity (77) from Theorem 12.1, we obtain the following behavioural differential equation:

behavioural differential equation	initial value
$\tau' = 1 + (\tau \otimes \tau)$	$\tau(0) = 0$

The Taylor series of $\tan(x)$ is notoriously difficult in that no closed formula for its elements, the so-called *tangent* numbers, is known. Here a representation for τ is constructed in the form of an infinite weighted automaton. Applying again the ‘splitting of derivatives’ procedure for the construction of a weighted automaton Q for τ , the first streams to be included as states of the automaton are τ , 1 , and $\tau \otimes \tau = \tau^2$. Computing the derivative of the latter, we find, using the differential equation for τ again,

$$\begin{aligned} (\tau^2)' &= 2\tau' \otimes \tau \\ &= 2(1 + \tau^2) \otimes \tau \\ &= 2\tau + 2\tau^3 \end{aligned}$$

Continuing this way, one obtains $Q = \{1, \tau, \tau^2, \tau^3, \tau^4, \dots\}$ with transitions and initial values as in the following diagram:



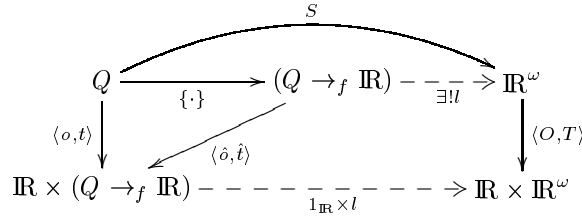
Thus we have obtained an, albeit infinite but extremely regular and simple weighted automaton, in which the state τ represents (itself, that is) the Taylor series of $\tan(x)$.

Applying Proposition 13.1 yields a closed formula for the n -th tangent number, formulated in terms of a finite sum over all paths of length n in the automaton that start in τ and end in 1. Possibly more interesting is the fact that the above automaton gives rise in a rather straightforward manner to the following continued fraction representation of the stream τ :

$$\tau = \frac{X}{1 - \frac{1 \cdot 2 \cdot X^2}{1 - \frac{2 \cdot 3 \cdot X^2}{1 - \frac{3 \cdot 4 \cdot X^2}{\ddots}}}}$$

A proof by coinduction is readily given, but rather we refer to Section 17, where a far more general result is proved.

We conclude the present section with a coalgebraic characterization of the assignment of streams $S(q)$ to the states q in a weighted stream automaton, which was defined above by means of a system of behavioural differential equations. (None of this will play a role in the remainder of the paper.) Consider a weighted stream automaton $(Q, \langle o, t \rangle)$. The following diagram tells it all:



The function $\{ \cdot \}$ is defined, for all $q, q' \in Q$, by

$$\{q\}(q') = \begin{cases} 1 & \text{if } q = q' \\ 0 & \text{otherwise} \end{cases}$$

For $\phi \in (Q \rightarrow_f \mathbb{R})$, with support $\{q_1, \dots, q_n\}$, the functions $\hat{o}$ and $\hat{t}$ are given by

$$\hat{o}(\phi) = o(q_1) \times \phi(q_1) + \dots + o(q_n) \times \phi(q_n)$$

and, for any $q \in Q$, by

$$\hat{t}(\phi)(q) = t(q_1)(q) \times \phi(q_1) + \dots + t(q_n)(q) \times \phi(q_n)$$

With these functions, we have provided $(Q \rightarrow_f \mathbb{R})$ with an ordinary (that is, as in Section 2) stream automaton structure $((Q \rightarrow_f \mathbb{R}), \langle \hat{o}, \hat{t} \rangle)$. It is consistent with the weighted automaton structure on Q , in that, for all $q \in Q$,

$$\hat{o}(\{q\}) = o(q), \quad \hat{t}(\{q\}) = t(q)$$

(We are dealing here with a generalisation of the ‘powerset’ construction from automata theory, which is used to transform nondeterministic automata into deterministic ones.) By the finality (Theorem 2.2) of the (ordinary) stream automaton $(\mathbb{R}^\omega, \langle O, T \rangle)$, there exists a unique homomorphism $l: (Q \rightarrow_f \mathbb{R}) \rightarrow \mathbb{R}^\omega$. The formal statement that the above diagram is meant to express, is that for all $q \in Q$,

$$S(q) = l(\{q\})$$

This can now be proved by a straightforward coinduction.

14 Multivariate streams

Recall from Theorem 4.3 that any stream $\sigma = (s_0, s_1, s_2, \dots)$ can be viewed as a power series in the formal variable X :

$$\sigma = s_0 + s_1 \times X^1 + s_2 \times X^2 + \dots$$

Another way of putting this is to view σ as a ‘weighted’ language over the one letter alphabet $\{X\}$, where the n -th element s_n of σ gives the multiplicity with which the word X^n ‘occurs’ in σ . This view on streams is in fact equivalent to the original definition of streams as functions $\sigma : \{0, 1, 2, \dots\} \rightarrow \mathbb{R}$, because of the existence of an isomorphism

$$\mathbb{R}^\omega \cong \mathbb{R}^{\{X\}^*}$$

where $\{X\}^* = \{\varepsilon, X, X^2, \dots\}$, the set of all finite words over the alphabet $\{X\}$ (with ε denoting the empty word).

Using this latter view, the present section generalises streams over the singleton alphabet $\{X\}$ to *multivariate* streams, which are ‘weighted’ languages over an arbitrary (possibly infinite) alphabet A . More precisely, the set of multivariate streams is defined by

$$\mathbb{R}^{A^*} = \{\sigma \mid \sigma : A^* \rightarrow \mathbb{R}\}$$

where A^* is the set of all finite words over the alphabet A . Multivariate streams are also known in the literature as *formal power series*:

$$\text{multivariate streams} \quad \equiv \quad \text{formal power series}$$

A multivariate stream σ assigns to each word $w \in A^*$ a real number $\sigma(w)$, which can be thought of as the weight or multiplicity with which this word ‘occurs’ in σ . We shall encounter denotations for multivariate streams that are similar to the ones we have seen before, such as

$$7 + 3X + \sqrt{2}(X \times Y) + Z^3 - \frac{1}{1 - (X \times Z)}$$

where the alphabet is $\{X, Y, Z\}$. As before, we develop a theory of multivariate streams in analogy to analysis (now involving the notion of partial stream derivative), but still more so than before, multivariate stream calculus will often behave differently from what we are used to. Most importantly, (convolution) product will no longer be commutative: in general,

$$X \times Y \neq Y \times X$$

In many cases, our guiding intuitions are to be taken from the theory of formal languages, rather than from analysis. The definition of the convolution product of multivariate streams, for instance, will be a generalisation of language *concatenation*.

The move from univariate streams to multivariate ones is conceptually dramatic and will lead to a far greater expressivity, but as we shall see, the technical development of the theory requires only minor variations on the results obtained thus far. Below we shall present multivariate streams in some more detail. The following sections will deal with multivariate stream calculus and some of its applications.

So let A be an arbitrary alphabet, the elements of which we call (formal) *variables*. The *initial value* of a multivariate stream $\sigma \in \mathbb{R}^{A^*}$ is denoted, as before, by $\sigma(0)$, and is defined as the value that σ assigns to the empty word $\varepsilon \in A^*$:

$$\sigma(0) = \sigma(\varepsilon)$$

For any variable $X \in A$, the *partial derivative* $\sigma_X \in \mathbb{R}^{A^*}$ of σ *with respect to* X , or X -derivative for short, is defined, for any $w \in A^*$, by

$$\sigma_X(w) = \sigma(X \cdot w)$$

(here $X \cdot w$ denotes concatenation of words).

Using the notions of initial value and partial derivative, the bit of universal coalgebra that we introduced for $\mathbb{R}^\omega$ in Section 2, can now be easily generalised as follows. A *bisimulation* on $\mathbb{R}^{A^*}$ is a relation $R \subseteq \mathbb{R}^{A^*} \times \mathbb{R}^{A^*}$ such that, for all σ and τ in $\mathbb{R}^{A^*}$,

$$\text{if } \sigma R \tau \text{ then } \begin{cases} \sigma(0) = \tau(0) & \text{and} \\ \sigma_X R \tau_X & \text{for all } X \in A \end{cases}$$

The greatest bisimulation relation $\sim$ is again called the *bisimilarity* relation.

Theorem 14.1 [Multivariate coinduction] *For all $\sigma, \tau \in \mathbb{R}^{A^*}$: $\sigma \sim \tau \Rightarrow \sigma = \tau$.*

Proof: Using the defining properties of a bisimulation relating σ and τ , one can prove for all words $w \in A^*$, by induction on their length, that $\sigma(w) = \tau(w)$. $\square$

Coinductive definitions will again be phrased in terms of behavioural differential equations, which are now *partial*. Unique existence of solutions of such equations will again follow from finality, which can be formulated using the following notion. A multivariate stream coalgebra or automaton is a pair $Q = (Q, \langle o, t \rangle)$ consisting of a set Q of states, together with an output function $o : Q \rightarrow \mathbb{R}$, and a transition function $t : Q \rightarrow Q^A$, with

$$Q^A = \{g \mid g : A \rightarrow Q\}$$

The transition function t assigns to each state q a function $t(q) : A \rightarrow Q$ that for each variable $X \in A$ yields a next state $t(q)(X)$. A homomorphism between stream automata $(Q, \langle o, t \rangle)$ and $(Q', \langle o', t' \rangle)$ is a function $f : Q \rightarrow Q'$ such that, for all q in Q and all $X \in A$, $o(q) = o'(f(q))$ and $f(t(q)(X)) = t'(f(q))(X)$:

$$\begin{array}{ccc} Q & \xrightarrow{f} & Q' \\ \langle o, t \rangle \downarrow & & \downarrow \langle o', t' \rangle \\ \mathbb{R} \times Q^A & \xrightarrow{1_{\mathbb{R}} \times f^A} & \mathbb{R} \times (Q')^A \end{array}$$

(The function $1_{\mathbb{R}} \times f^A$ maps a pair $\langle r, g \rangle \in \mathbb{R} \times Q^A$ to $\langle r, f \circ g \rangle \in \mathbb{R} \times (Q')^A$.) The set $\mathbb{R}^{A^*}$ of all multivariate streams can itself be turned into such an automaton as well. Defining $O : \mathbb{R}^{A^*} \rightarrow \mathbb{R}$ by $O(\sigma) = \sigma(0)$ and $T : \mathbb{R}^{A^*} \rightarrow (\mathbb{R}^{A^*})^A$ by $T(\sigma)(X) = \sigma_X$, for all $X \in A$, we obtain a multivariate stream automaton $(\mathbb{R}^{A^*}, \langle O, T \rangle)$, which, as in the case of $\mathbb{R}^\omega$, is final.

Theorem 14.2 *For any multivariate automaton $(Q, \langle o, t \rangle)$ there exists a unique homomorphism $l : Q \rightarrow \mathbb{R}^{A^*}$:*

$$\begin{array}{ccc} Q & \xrightarrow{\exists! l} & \mathbb{R}^{A^*} \\ \langle o, t \rangle \downarrow & & \downarrow \langle O, T \rangle \\ \mathbb{R} \times Q^A & \xrightarrow{1_{\mathbb{R}} \times l^A} & \mathbb{R} \times (\mathbb{R}^{A^*})^A \end{array}$$

Proof: The function $l : Q \rightarrow \mathbb{R}^{A^*}$ assigns to a state q in Q the multivariate stream $l(q)$ defined, for any variable $X \in A$ and word $w \in A^*$, using induction on the length of words, by

$$l(q)(\varepsilon) = o(q), \quad l(q)(X \cdot w) = l(t(q)(X))(w)$$

This function is a homomorphism from Q to $\mathbb{R}^{A^*}$ and, by coinduction, the only such one. $\square$

15 Multivariate stream calculus

Almost literally following the calculus that we have developed for (univariate) streams, a calculus for multivariate streams is presented. Operators will again be defined by (systems of) behavioural differential equations, which are now partial. We shall not prove the existence of unique solutions for such equations; it follows from the finality of $\mathbb{R}^{A^*}$ in very much the same way as before (cf. [Rut99, Rut00a]).

We present all the operators in one go, by the following large system of behavioural partial differential equations. For all $X, Y \in A$, $\sigma, \tau \in \mathbb{R}^{A^*}$, $r \in \mathbb{R}$, the following equations have unique solutions:

behavioural differential equation	initial value	name
$r_X = 0$	$r(0) = r$	constant
$Y_Y = 1, Y_X = 0 \ (Y \neq X)$	$Y(0) = 0$	(formal) variable
$(\sigma + \tau)_X = \sigma_X + \tau_X$	$(\sigma + \tau)(0) = \sigma(0) + \tau(0)$	sum
$(\sigma \times \tau)_X = (\sigma_X \times \tau) + (\sigma(0) \times \tau_X)$	$(\sigma \times \tau)(0) = \sigma(0) \times \tau(0)$	(convolution) product
$(\sigma^{-1})_X = -\sigma(0)^{-1} \times (\sigma_X \times \sigma^{-1})$	$(\sigma^{-1})(0) = \sigma(0)^{-1}$	inverse ($\sigma(0) \neq 0$)
$(\sqrt{\sigma})_X = \sigma_X \times (\sqrt{\sigma(0)} + \sqrt{\sigma})^{-1}$	$\sqrt{\sigma}(0) = \sqrt{\sigma(0)}$	square root ($\sigma(0) > 0$)
$(\sigma \otimes \tau)_X = (\sigma_X \otimes \tau) + (\sigma \otimes \tau_X)$	$(\sigma \otimes \tau)(0) = \sigma(0) \times \tau(0)$	shuffle product
$(\sigma^{-1})_X = -\sigma_X \otimes (\sigma^{-1} \otimes \sigma^{-1})$	$\sigma^{-1}(0) = \sigma(0)^{-1}$	shuffle inverse ($\sigma(0) \neq 0$)
$\exp(\sigma)_X = \exp(\sigma) \otimes \sigma_X$	$\exp(\sigma)(0) = e^{\sigma(0)}$	exponentiation

Almost all of the above definitions are obtained by replacing in the corresponding equation for streams the occurrences of $'$ (stream derivative) by X -derivatives. The only exception is the variable Y , for which it makes a difference whether one takes a Y -derivative (which yields 1) or an X -derivative (which yields 0 for $X \neq Y$). Furthermore, basic properties such as associativity, commutativity, and distributivity, hold pretty much as before; we will not repeat them here. There is, however, one notable exception. As mentioned earlier, convolution product is no longer commutative. Although for $r \in \mathbb{R}$ and $\sigma \in \mathbb{R}^{A^*}$, one still has

$$r \times \sigma = \sigma \times r$$

this no longer applies to arbitray τ ; generally,

$$\sigma \times \tau \neq \tau \times \sigma$$

Indeed, convolution product is more like language concatenation, as expressed by the following formula, with which convolution product of formal power series is traditionally introduced: for all $w \in A^*$,

$$(\sigma \times \tau)(w) = \sum_{u, v: w=u \cdot v} \sigma(u) \times \tau(v)$$

It generalises identity(5) and shares its disadvantage for actual computation, of being an indexed sum. As always, our reasoning will be coinductive and therefore, we shall not need it.

Note that since product is no longer commutative, we have to be more careful in our notational conventions. We define

$$\frac{\sigma}{\tau} \equiv \sigma \times \tau^{-1}$$

which generally is different from $\tau^{-1} \times \sigma$. Further note that the operation of (convolution) inverse still is both a left and a right inverse:

$$\begin{aligned} \sigma \times \sigma^{-1} &= 1 \\ \sigma^{-1} \times \sigma &= 1 \end{aligned}$$

Proofs of the above and similar such identities can be given using coinduction Theorem 14.1. Also the notion of bisimulation-up-to and the corresponding proof principle of coinduction-up-to (Theorem 4.2) can be straightforwardly adapted to the multivariate case.

Many behavioural partial differential equations can again be solved algebraically, as in Section 5, now using the following generalisation of Theorem 4.1, which can be easily proved by coinduction.

Theorem 15.1 [Fundamental Theorem] *For all $\sigma \in \mathbb{R}^{A^*}$,*

$$\sigma = \sigma(0) + \sum_{X \in A} X \times \sigma_X$$

By way of example, Theorem 15.1 is used to solve the following partial differential equation:

behavioural differential equation	
$\sigma_X = (Y \times \sigma) + 2$	X-derivative
$\sigma_Y = \sigma$	Y-derivative
$\sigma(0) = 1$	initial value

By Theorem 15.1, we have

$$\begin{aligned} \sigma &= \sigma(0) + (X \times \sigma_X) + (Y \times \sigma_Y) \\ &= 1 + X \times ((Y \times \sigma) + 2) + Y \times \sigma \\ &= 1 + 2X + (XY + Y) \times \sigma \end{aligned}$$

As a consequence, $(1 - (XY + Y)) \times \sigma = 1 + 2X$, whence

$$\sigma = (1 - (XY + Y))^{-1} (1 + 2X)$$

(Note that the order of multiplication is important here: $\tau = (1 + 2X)(1 - (XY + Y))^{-1}$ is not a solution of the same equation. For one thing, $\tau_Y \neq \tau$.)

Many of the earlier identities for streams also apply to multivariate streams. For instance,

$$\begin{aligned} \exp(\sigma) &= 1 + \frac{\sigma^1}{1!} + \frac{\sigma^2}{2!} + \dots \\ \exp(\sigma) \otimes \exp(\tau) &= \exp(\sigma + \tau) \\ \frac{1}{1 - rX} \otimes \frac{1}{1 - sY} &= \frac{1}{1 - (rX + sY)} \end{aligned}$$

16 Weighted automata for multivariate streams

In preparation of the applications of multivariate stream calculus, this section shows how the definition of weighted stream automaton can be adapted to multivariate streams.

A *multivariate $\mathbb{R}$ -weighted stream automaton*, or *multivariate weighted automaton* for short, is a pair $Q = (Q, \langle o, t \rangle)$ consisting of a set Q of states, together with an output function $o : Q \rightarrow \mathbb{R}$, and a transition function $t : Q \rightarrow (Q \rightarrow_f \mathbb{R})^A$, where, as before, $(Q \rightarrow_f \mathbb{R})$ contains only functions of finite support. The output function o assigns to each state q in Q a real number $o(q)$ in $\mathbb{R}$. The transition function t assigns to a state q in Q a function $t(q) : (Q \rightarrow_f \mathbb{R})^A$, which specifies for any variable $X \in A$ and state q' in Q a real number $t(q)(X)(q')$ in $\mathbb{R}$. As before, this number can be thought of as the weight (cost, multiplicity, duration, etc.) with which the (X) -transition from q to q' occurs. The following notation will be used:

$$\begin{aligned} q &\xrightarrow{X|r} q' \text{ iff } t(q)(X)(q') = r \\ q &\xrightarrow{r} \text{ iff } o(q) = r \end{aligned}$$

which will again allow us to present weighted multivariate automata by pictures. Note that multivariate weighted automata are nondeterministic in that we can have transitions

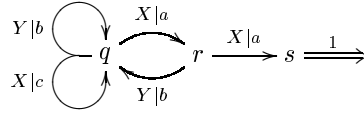
$$q \xrightarrow{X|r_1} q_1, \quad q \xrightarrow{X|r_2} q_2$$

for different states $q_1 \neq q_2$.

The behaviour of a state q in a multivariate weighted automaton Q is now a multivariate stream $S(q) \in \mathbb{R}^{A^*}$, defined coinductively as follows. Let $X \in A$, let $\{q_1, \dots, q_n\}$ be the support of $t(q)(X)$, and let $r_i = t(q)(X)(q_i)$, for $1 \leq i \leq n$. Then $S(q)$ is defined by the following system of partial differential equations (one for each $q \in Q$ and $X \in A$):

behavioural differential equation	
$S(q)_X = r_1 S(q_1) + \dots + r_n S(q_n)$	X -derivative
$\dots$	$\dots$
$S(q)(0) = o(q)$	initial value

For an example, consider the following finite automaton over the alphabet $\{X, Y\}$, with states $\{q, r, s\}$ and weights $a, b, c \in \mathbb{R}$:



Its behaviour is defined by the following system of partial differential equations:

behavioural differential equations	
$S(q)_X = (a \times S(r)) + (c \times S(q))$	X -derivative
$S(q)_Y = b \times S(q)$	Y -derivative
$S(q)(0) = 0$	initial value
$S(r)_X = a \times S(s)$	X -derivative
$S(r)_Y = b \times S(q)$	Y -derivative
$S(r)(0) = 0$	initial value
$S(s)_X = 0$	X -derivative
$S(s)_Y = 0$	Y -derivative
$S(s)(0) = 1$	initial value

For $S(q)$ the following expression can be obtained

$$S(q) = (1 - abXY - cX - bY)^{-1} a^2 X^2$$

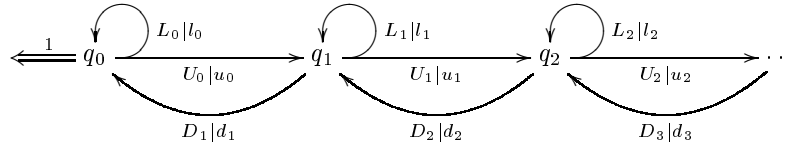
by solving the system of equations as before, using Theorem 15.1.

Propositions 13.1 and 13.2, and Theorems 13.3 and 13.4 can be straightforwardly generalised to multivariate weighted automata. Since they will play no role in the applications presented in Section 17, they are omitted.

17 Application: coinductive counting

Multivariate weighted automata turn out to be very well suited for the modelling of various classical counting problems from combinatorics. In this section, the techniques from the previous sections are illustrated by means of a number of examples of what could be called *coinductive counting*.

Consider the following infinite multivariate weighted automaton



with state space $\{q_0, q_1, q_2, \dots\}$, weights $l_i, u_i, d_i \in \mathbb{R}$, and alphabet

$$\{L_0, L_1, L_2, \dots\} \cup \{U_0, U_1, U_2, \dots\} \cup \{D_1, D_2, D_3, \dots\}$$

If we picture this automaton vertically, with q_0 at the bottom, we may think of the symbols L_i , U_i and D_i as transition steps (starting in the state q_i and) representing moves that remain *level*, go *up*, or go *down*, respectively. The multivariate stream behaviour of this automaton is characterised by the following theorem, which is closely related to [Fla80] (see also Section 18).

Theorem 17.1 *The multivariate stream behaviour of the state q_0 in the automaton above is given by the following continued fraction:*

$$S(q_0) = \frac{1}{1 - (l_0 \times L_0) - \frac{(u_0 \times U_0) \mid (d_1 \times D_1)}{1 - (l_1 \times L_1) - \frac{(u_1 \times U_1) \mid (d_2 \times D_2)}{1 - (l_2 \times L_2) - \frac{(u_2 \times U_2) \mid (d_3 \times D_3)}{\ddots}}}}$$

where the following notational convention is used: for any $\sigma, \tau, \rho \in \mathbb{R}^{A^*}$,

$$\frac{\sigma \mid \tau}{\rho} \equiv \sigma \times \rho^{-1} \times \tau$$

Before we prove this equality, let us make sure we understand how the continued fraction on the right is actually defined. *Suppose* for a second that the continued fraction is well-defined and that, more generally, the multivariate streams σ_i below are well-defined, for any $i \geq 0$:

$$\sigma_i = \frac{1}{1 - (l_i \times L_i) - \frac{(u_i \times U_i) \mid (d_{i+1} \times D_{i+1})}{1 - (l_{i+1} \times L_{i+1}) - \frac{(u_{i+1} \times U_{i+1}) \mid (d_{i+2} \times D_{i+2})}{\ddots}}}$$

For such multivariate streams σ_i we would have, for any $i \geq 0$,

$$\sigma_i = \frac{1}{1 - (l_i \times L_i) - (u_i \times U_i) \times \sigma_{i+1} \times (d_{i+1} \times D_{i+1})}$$

The latter equations can now simply be taken as a definition. Or, rather more precisely, from these equations we can now read off the following system of partial behavioural differential equations, the unique solution of which is taken as the formal definition of the multivariate streams σ_i :

behavioural differential equation	
$(\sigma_i)_{L_i} = l_i \times \sigma_i$	L_i -derivative
$(\sigma_i)_{U_i} = u_i \times \sigma_{i+1} \times (d_{i+1} \times D_{i+1}) \times \sigma_i$	U_i -derivative
$(\sigma_i)_X = 0$	X -derivative, all $X \neq L_i, U_i$
$\sigma_i(0) = 1$	initial value

Proof: A proof of Theorem 17.1 on the basis of the above system of differential equations formally defining the streams σ_i , is now a straightforward coinduction. Let $\tau_0 = \sigma_0$ and let, for $i \geq 1$,

$$\tau_i = \sigma_i \times (d_i \times D_i) \times \sigma_{i-1} \times (d_{i-1} \times D_{i-1}) \times \cdots \times \sigma_1 \times (d_1 \times D_1) \times \sigma_0$$

Then $S(q_0) = \sigma_0$ follows by coinduction (Theorem 14.1) from the fact that

$$\{\langle r \times S(q_i), r \times \tau_i \rangle \mid r \in \mathbb{R}, i \geq 0\}$$

is a bisimulation relation on $\mathbb{R}^{A^*}$. □

Many counting problems can be expressed in terms of the weighted automaton above, which is illustrated by the following example. As an alphabet we take the set consisting of the two bracket symbols

$$A = \{ <, > \}$$

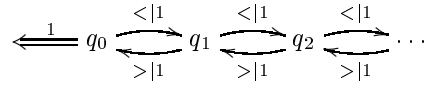
We are interested in counting the number of words over this alphabet that are “well-bracketed” in the obvious sense: the number of opening brackets used should equal the number of closing brackets, and any prefix of the word should contain at least as many opening as closing brackets. For instance, here are all the well-bracketed words containing three opening and three closing brackets:

$$<<<>>>, <<><>>, <<>><>, <><<>>, <><><>$$

The following (classical) automaton, with q_0 as initial and only accepting state, is a representation of the set of all well-bracketed words, which correspond precisely to all the paths starting and ending in the state q_0 :



Adding everywhere the trivial weight 1, and giving q_0 the output value 1 (and all other states output value 0), we obtain the following multivariate weighted automaton



which clearly is an instantiation of our automaton above, with $U_i = <$, $D_i = >$, $u_i = d_i = 1$ (and $l_i = 0$). By Theorem 17.1, we have

$$S(q_0) = \frac{1}{1 - \frac{<|>}{1 - \frac{<|>}{1 - \frac{<|>}{\ddots}}}}$$

Putting $\sigma = S(q_0)$, one finds

$$\sigma = \frac{1}{1 - < \times \sigma \times >}$$

which is equivalent to

$$\sigma = 1 + < \times \sigma \times > \times \sigma \tag{87}$$

Recall that the problem we are trying to solve is that of counting the respective numbers of well-bracketed words. The (univariate) stream of all these numbers is obtained from the multivariate stream σ by identifying the variables $<$ and $>$, replacing them by one and the same variable X :

$$\beta = \sigma(X, X)$$

(Although we have not introduced a general operation of substitution for multivariate streams, this type of variable substitution can be readily defined formally by means of behavioural differential equations.) As an immediate consequence of (87), β is the unique stream satisfying the following equation:

$$\beta = 1 + (X \times \beta \times X \times \beta)$$

Since β is now a univariate stream, that is, an element of $\mathbb{R}^\omega$, the techniques of Section 7 apply, and we find

$$\begin{aligned}\beta &= \frac{1}{1 - \frac{X^2}{1 - \frac{X^2}{1 - \frac{X^2}{\ddots}}}} \\ &= \frac{2}{1 + \sqrt{1 - 4X^2}} \\ &= (1, 0, 1, 0, 2, 0, 5, 0, 14, 0, \dots)\end{aligned}$$

Here we recognise the numbers at the even positions as the Catalan numbers, which we already encountered as the solution of equation (35). (As an aside, note that, indeed,

$$\frac{2}{1 + \sqrt{1 - 4X^2}} = \frac{2}{1 + \sqrt{1 - 4X}}(X^2)$$

where $(\dots)(X^2)$ refers to the composition operation on streams, as in identity (26).)

In the above example, the weights in the automaton, all being 1, played only a trivial role. In conclusion of this section, we recall the weighted automaton for the Taylor series of the tangent function from Section 13, which has more interesting weights. In the notation of the present section, it looks like

$$\begin{array}{ccccccc} \xleftarrow{1} & q_0 & \xleftarrow{X|1} & q_1 & \xrightleftharpoons[X|2]{X|1} & q_2 & \xrightleftharpoons[X|3]{X|2} & q_3 & \xrightleftharpoons[X|4]{X|3} & \dots \end{array}$$

where now the state q_1 represent the Taylor series τ of $\tan(x)$, that is, $\tau = S(q_1)$. By (the proof of) Theorem 17.1, we find that

$$\tau = \frac{X}{1 - \frac{1 \cdot 2 \cdot X^2}{1 - \frac{2 \cdot 3 \cdot X^2}{1 - \frac{3 \cdot 4 \cdot X^2}{\ddots}}}}$$

which was mentioned but not yet proved in Section 13.

18 Discussion and related work

General references on (universal) coalgebra are [JR97, Rut00b]. Earlier work on a coalgebraic approach to automata, formal languages, and formal power series include [Rut98, Rut99]. The formulation of a coinductive proof principle in terms of bisimulation relations goes back to work by Aczel and Mendler [Acz88, AM89], generalising Park and Milner's [Par81, Mil80] notion of bisimulation to a categorical setting. The notion of stream derivative is a special instance of the notion of *input derivative*, which goes back to [Brz64]. It plays a role also in [Con71], where the chapter 'The differential calculus of events' already suggests a connection with classical calculus.

The definition of bisimulation-up-to is a variation on a similar notion by Milner [Mil89] (see also [San98]). In [Bar00], variations and coalgebraic generalisations of coinductive proof methods are given. The present notion of bisimulation-up-to (identity and sum), can be easily generalised along the lines of [Bar00], to a version which would allow derivatives to be bisimilar up to arbitrary contexts (including product, inverse, and the other operators). The proofs of for instance identities (15) and (25) could be simplified if we would have used bisimulation-up-to product.

The present paper extends [Rut00a], repeating part of its basic definitions and results on streams. A number of new operators have been added, as well as many new identities (including those on exponentiation and shuffle elimination). Moreover, all of the applications are new.

Pavlović and Escardó’s paper on calculus in coinductive form [PE98], emphasising the close connection between classical analysis and coinduction, has been an important source of inspiration for our work, motivating in particular the application of stream calculus to analytical differential equations. Other than parts of Theorems 10.1 and 12.1, the papers have, technically speaking, not very much in common.

Motivating sources of examples of streams and stream operators have been the books [GKP94] and [Wil94], and the papers by McIlroy [McI99, McI01] and Karczmarszuk [Kar97, Kar00].

The solution of difference equations by means of stream calculus is conceptually very simple, since the entire game is played within the world of streams. In contrast to the classical technique of generating functions (as in [GKP94] and [Wil94]), functions (from $\mathbb{R}$ to $\mathbb{R}$) are just not needed in stream calculus, and convergence issues thereby simply do not enter the picture.

As we have seen, streams can be viewed as formal power series (in one variable), which are often used as a formal alternative to generating functions, precisely to avoid convergence considerations. We see some advantages of stream calculus also over the use of formal power series. First, there is the rigorous use of coinduction, both in definitions and in proofs, which makes stream calculus possibly more formal than the use of ‘formal’ power series usually is. Notably this applies to the use of the operation of inverse, which is not always treated strictly formally within theories of power series. Secondly, stream calculus is a more expressive calculus, because of the simultaneous presence of two types of multiplication: convolution product and shuffle product; of two corresponding types of inverse; and of two types of derivatives (ordinary and analytical). This is for instance illustrated by the example of a divergent recurrence in Section 9. The interplay between the various operators, notably between convolution inverse and shuffle product, moreover gives rise to quite a few interesting identities (such as in Theorem 10.2), which seem to be the underlying facts for various results in different parts of mathematics. For instance, identity (68) is used both in the proof of the Euler formula (in Section 11) and in the proof of the characterisation of the Taylor transform of products with the exponential function (identity (85)).

Also in our proof of the Euler formula, no assumptions on convergence need to be made, in contrast to certain analytical proofs (as in [Sch68, 11.38]). It shares with certain proofs in operator arithmetic the fact that it is very short and transparent, but has the additional advantage of being entirely formal, whereas the latter proofs often are by “a somewhat optimistic application of operator arithmetic” (cf. [Sch68, page 75]).

Solving differential equations in stream calculus essentially amounts to the classical method of undetermined coefficients [BR78, p.82], with the difference that the obtained difference equations for the Taylor coefficients of the analytical solution, are solved within the world of streams. This approach is technically closely related to the use of Laplace transforms (cf. [Sne72, Mik83]), because the operation of assigning the Taylor series $\mathcal{T}(f)$ to an analytical function f implicitly uses the Laplace-Carson transform (introduced in Section 10). Formulae such as (80)-(86) are similar to but different from the formulae of the corresponding Laplace transforms (cf. [Sne72, page 519]). Conceptually, the use of stream calculus is different and, again, a bit simpler than these traditional approaches. In particular, analytical integration plays no role, since the difference equations are solved by stream integration (applying the Fundamental Theorem of stream calculus, Theorem 4.1).

For a general reference on (rational) formal power series, see [BR88]. Theorems 13.3 and 13.4 are classical results. New about our use of weighted automata as representations for streams are: -the coinductive definition of their behaviour; -the way such automata are constructed by means of splitting derivatives (as in the proof of Theorem 13.3); -and our use of infinite weighted automata (as in the example of $\mathcal{T}(\tan)$). Part of all this can already be found in [Rut00a]. New with respect to the latter is the use of continued fractions, and the application in Section 17 to coinductive counting. Theorem 17.1 is a (slightly more general) variation on a classical result by Flajolet [Fla80, Theorem 1], phrased here in terms of weighted automata and with a somewhat simpler coinductive proof. Weighted automata seem to offer an interesting alternative to the structures for

modelling enumerative problems used in [Fla80], such as sets of labelled paths and path diagrams.

Another approach to counting is the categorical theory of species [BLL98]. It offers a framework that is far more general than the present calculus of streams, but there are many connections. It would be worthwhile, more generally speaking, to investigate the possible role of coinduction in the world of species in some detail.

In conclusion, we observe that there are some obvious ways in which the results of stream calculus can be generalised. For one thing, one can look at other fields (such as the complex numbers) or even arbitrary semirings (such as the Booleans or so-called tropical semirings [Gun98]). The role of inverse will then be replaced by the operation of Kleene star. Another variation would be to consider other structures such as binary trees. Finally, it could be interesting to look at partial streams, which would correspond to the elements of the final coalgebra of type $\mathbb{R} \times (1 + (-))$.

19 Appendix

Here is a more general theorem on the unique existence of the solution of systems of behavioural differential equations. Let $\Sigma = \{f, g, \dots\}$ be a set of function symbols with arities $r_f, r_g, \dots$, and let T be the set of all terms built from symbols in Σ and the elements in $\mathbb{R}^\omega$ (now viewed as variables). Consider for each $f \in \Sigma$ with arity $r = r_f$ a term $t_f \in T$ containing (at most) the variables $\sigma_1, \dots, \sigma_r, \sigma'_1, \dots, \sigma'_r$, and $\sigma_1(0), \dots, \sigma_r(0)$ (the latter real numbers considered as constant streams). And consider a real-valued function $h_f : \mathbb{R}^r \rightarrow \mathbb{R}$.

Theorem 19.1 *The following system of behavioural differential equations (one for each $f \in \Sigma$):*

<i>behavioural differential equation</i>	<i>initial value</i>
$f(\sigma_1, \dots, \sigma_r)' = t_f$	$f(\sigma_1, \dots, \sigma_r)(0) = h_f(\sigma_1(0), \dots, \sigma_r(0))$

has a unique solution. That is, there exists for each $f \in \Sigma$ a unique stream operator (denoted by the same symbol) $f : (\mathbb{R}^\omega)^r \rightarrow \mathbb{R}^\omega$ satisfying the equation above.

Most of the equations that occur in Section 4 fit into the format of the theorem. For instance, in

behavioural differential equation	initial value	name
$(\sigma \times \tau)' = (\sigma' \times \tau) + (\sigma(0) \times \tau')$	$(\sigma \times \tau)(0) = \sigma(0) \times \tau(0)$	(convolution) product

we have $f = \times$, $r = 2$, $t_f = (\sigma' \times \tau) + (\sigma(0) \times \tau')$ with variables $\tau, \sigma', \tau', \sigma(0)$, and h_f is the multiplication of real numbers. The only operator that does not quite fit into the format is the generalised sum (because it takes a set I many arguments). It is, however, rather easy either to generalise the formulation of the present theorem, or to treat the case of the generalised sum separately.

Proof: The set of all terms T can be turned into a stream coalgebra, by induction on the syntactic complexity of the terms, and following the equations of the theorem. By finality, there exists a unique homomorphism from the coalgebra T into the coalgebra $\mathbb{R}^\omega$, which assigns to each *syntactic term* $f(\sigma_1, \dots, \sigma_r)$ a stream in $\mathbb{R}^\omega$. This stream is then what we define to be the effect of the *operator* f on the argument streams $(\sigma_1, \dots, \sigma_r)$. Rather than providing more detail here, we refer the reader to [Rut00a], where this construction is carried out in all detail for (a subset of) the operators of stream calculus. For a more general treatment of this type of coinductive definitions, see also [Bar00]. $\square$

Acknowledgements

I am much indebted to Alexandru Baltag, Falk Bartels, and Alexander Kurz for numerous stimulating discussions, suggestions, and corrections. Special thanks are due to Falk Bartels for detailed comments on a draft of this paper. I am also grateful to the following persons for various types of comments, suggestions, and pointers to the literature: Jan Komenda, Gilbert Labelle, Doug

McIlroy, Dusko Pavlovic, Andy Pitts, Jan van Schuppen, Nico Temme, and the members of the ACG Colloquium, headed by Jaco de Bakker.

References

- [Acz88] P. Aczel. *Non-well-founded sets*. Number 14 in CSLI Lecture Notes. Center for the Study of Languages and Information, Stanford, 1988.
- [AM89] P. Aczel and N. Mendler. A final coalgebra theorem. In D.H. Pitt, D.E. Ryehard, P. Dybjer, A. M. Pitts, and A. Poigne, editors, *Proceedings category theory and computer science*, number 389 in Lecture Notes in Computer Science, pages 357–365, 1989.
- [Bar00] F. Bartels. Generalised coinduction. Report SEN-R0043, CWI, 2000. Available at URL www.cwi.nl. Extended abstract in *Electronic Notes in Theoretical Computer Science* Vol.44.1, Elsevier, 2001.
- [BLL98] F. Bergeron, G. Labelle, and P. Leroux. *Combinatorial species and tree-like structures*, volume 67 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, 1998.
- [BR78] G. Birkhoff and G.-C. Rota. *Ordinary differential equations (third edition)*. John Wiley and Sons, 1978.
- [BR88] J. Berstel and C. Reutenauer. *Rational series and their languages*, volume 12 of *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, 1988.
- [Brz64] J.A. Brzozowski. Derivatives of regular expressions. *Journal of the ACM*, 11(4):481–494, 1964.
- [Com74] L. Comtet. *Advanced combinatorics*. D. Reidel Publishing Company, 1974.
- [Con71] J.H. Conway. *Regular algebra and finite machines*. Chapman and Hall, 1971.
- [Fla80] P. Flajolet. Combinatorial aspects of continued fractions. *Discrete Mathematics*, 32:125–161, 1980.
- [GKP94] R.L. Graham, D.E. Knuth, and O. Patashnik. *Concrete mathematics, second edition*. Addison-Wesley, 1994.
- [Gun98] J. Gunawardena. *Idempotency*. Publications of the Newton Institute. Cambridge University Press, 1998.
- [JR97] Bart Jacobs and Jan Rutten. A tutorial on (co)algebras and (co)induction. *Bulletin of the EATCS*, 62:222–259, 1997. Available at URL: www.cwi.nl/~janr.
- [Kar97] J. Karczmarszuk. Generating power of lazy semantics. *Theoretical Computer Science*, 187:203–219, 1997.
- [Kar00] J. Karczmarszuk. Lazy processing and optimization of discrete sequences. In *Proceedings of the JFLA'2000*, 2000. In French.
- [McI99] M.D. McIlroy. Power series, power serious. *Journal of Functional Programming*, 9:323–335, 1999.
- [McI01] M.D. McIlroy. The music of streams. *Information Processing Letters*, 77:189–195, 2001.
- [Mik83] J. Mikusinski. *Operational calculus*, volume 109. Pergamon press, 1983.
- [Mil80] R. Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, 1980.

- [Mil89] R. Milner. *Communication and Concurrency*. Prentice Hall, 1989.
- [Par81] D.M.R. Park. Concurrency and automata on infinite sequences. In P. Deussen, editor, *Proceedings 5th GI conference*, volume 104 of *Lecture Notes in Computer Science*, pages 15–32. Springer-Verlag, 1981.
- [PE98] D. Pavlović and M. Escardó. Calculus in coinductive form. In *Proceedings of the 13th Annual IEEE Symposium on Logic in Computer Science*, pages 408–417. IEEE Computer Society Press, 1998.
- [Rut98] J.J.M.M. Rutten. Automata and coinduction (an exercise in coalgebra). Report SEN-R9803, CWI, 1998. Available at URL: www.cwi.nl. Also in the proceedings of CONCUR '98, LNCS 1466, 1998, pp. 194–218.
- [Rut99] J.J.M.M. Rutten. Automata, power series, and coinduction: taking input derivatives seriously (extended abstract). Report SEN-R9901, CWI, 1999. Available at URL: www.cwi.nl. Also in the proceedings of ICALP '99, LNCS 1644, 1999, pp. 645–654.
- [Rut00a] J.J.M.M. Rutten. Behavioural differential equations: a coinductive calculus of streams, automata, and power series. Report SEN-R0023, CWI, 2000. Available at URL: www.cwi.nl.
- [Rut00b] J.J.M.M. Rutten. Universal coalgebra: a theory of systems. *Theoretical Computer Science*, 249(1):3–80, 2000.
- [San98] D. Sangiorgi. On the bisimulation proof method. *Mathematical Structures in Computer Science*, 8(5):447–479, 1998.
- [Sch68] F. Scheid. *Theory and problems of numerical analysis (Schaum's outline series)*. McGraw-Hill, 1968.
- [Sne72] I.N. Sneddon. *The use of integral transforms*. McGraw-Hill, 1972.
- [Wil94] H.S. Wilf. *Generatingfunctionology*. Academic Press, San Diego, CA, 1994.