A coinductive calculus of streams

J.J.M.M. RUTTEN

Mathematical Structures in Computer Science / Volume 15 / Issue 01 / February 2005, pp 93 - 147
DOI: 10.1017/S0960129504004517, Published online: 28 January 2005

Link to this article: http://journals.cambridge.org/abstract_S0960129504004517

How to cite this article:
J.J.M.M. RUTTEN (2005). A coinductive calculus of streams. Mathematical Structures in Computer Science, 15, pp 93-147 doi:10.1017/S0960129504004517

Math. Struct. in Comp. Science (2005), *vol.* 15, *pp.* 93–147. © 2005 Cambridge University Press
DOI: 10.1017/S0960129504004517 Printed in the United Kingdom

A coinductive calculus of streams

J.J.M.M. RUTTEN

CWI and VUA, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands
Email: janr@cwi.nl, *URL:* www.cwi.nl/∼janr

Received 20 February 2002; revised 20 January 2004

We develop a coinductive calculus of streams based on the presence of a final coalgebra structure on the set of streams (infinite sequences of real numbers). The main ingredient is the notion of stream derivative, which can be used to formulate both coinductive proofs and definitions. 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.

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.

The 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:

$$\langle O, T \rangle : \mathbb{R}^\omega \to \mathbb{R} \times \mathbb{R}^\omega, \quad \sigma \mapsto \langle \sigma(0), \sigma' \rangle.$$

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

$$\sigma(0) = s_0, \quad \sigma' = (s_1, s_2, s_3, \ldots).$$

(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*, which are 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, we introduce many stream operators: 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
$\Sigma_{n=0}^{\infty} \sigma_n$	infinite sum
σ^{-1}	(multiplicative) inverse
$\sigma \circ \tau$	stream composition
$\sqrt{\sigma}$	square root
$\sigma \otimes \tau$	shuffle product
$\sigma^{\underline{-1}}$	shuffle inverse
$\sigma^{\underline{1/2}}$	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 then rather more advanced facts are proved. Among the most interesting ones are the identities for stream exponentiation and 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, which relate 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);
— Combinatorics: coinductive counting (Section 14).

The present paper can be seen within the ongoing work on coalgebra as a study in *concrete* coalgebra, as opposed to the more general and abstract studies of *universal coalgebra*. Another way of putting this 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 introduced explicitly.

To conclude this introduction, we briefly summarise related work, which will be discussed more extensively in Section 15:

— The present paper builds on Rutten (2000a), by repeating some of its basics, extending its results on streams, and adding the applications mentioned above.
— An important source of inspiration has been Pavlović and Escardó (1998), 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; Wilf (1994) and Aczel (1988) 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 (Berstel and Reutenauer 1988) are often taken as the more formal version of generating functions. 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. Finally, the classical use of (generating functions and) formal power series as representations of streams, distinguishes between (at least) two different types of series, namely normal type and exponential type (the difference consists of the presence of a factorial number). In stream calculus, both representations (and the various operations corresponding to them) can be dealt with in a uniform way, and inside one and the same framework. In other words, the need for the distinction disappears.

— 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 McIlroy (1999; 2001) and Karczmarczuk (1997; 2000) 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 in this section. 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 used in Section 3 as the basis for coinductive definition schemata.

The set of all streams is formally defined by

$$\mathbb{R}^\omega = \{\sigma \mid \sigma : \{0, 1, 2, \ldots\} \to \mathbb{R}\}.$$

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

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

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

$$\sigma' = (s_1, s_2, s_3, \ldots).$$

For any $n \geqslant 0$, 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 $k \geqslant 0$, 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

$$\forall\, n \geqslant 0,\; \sigma(n) \; = \; \tau(n). \tag{2}$$

What general methods are there for establishing the validity of (2)? 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 IR^ω, 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 IR^ω is a relation $R \subseteq \mathrm{IR}^\omega \times \mathrm{IR}^\omega$ such that, for all σ and τ in IR^ω,

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

(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 again bisimulations. 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, thus, the greatest bisimulation.

Theorem 2.1 (Coinduction). If two streams σ and τ are bisimilar ($\sigma \sim \tau$), it follows that $\sigma(n) = \tau(n)$, for all $n \geqslant 0$, and, consequently, $\sigma = \tau$. That is, for all $\sigma, \tau \in \mathrm{IR}^\omega$,

$$\sigma \sim \tau \Rightarrow \; \sigma = \tau.$$

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

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

We shall see many examples of proofs by coinduction: one of the main reasons we became interested in $\mathbb{R}^\omega$ 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 $o : Q \to \mathbb{R}$, and a transition function $t : Q \to Q$. A homomorphism between stream automata $(Q_1, \langle o_1, t_1\rangle)$ and $(Q_2, \langle o_2, t_2\rangle)$ is a function $f : Q_1 \to Q_2$ such that, for all q in Q_1, $o_1(q) = o_2(f(q))$ and $f(t_1(q)) = t_2(f(q))$ or, in other words, such that the following diagram commutes:

$$\begin{tikzcd} Q_1 \arrow[r, "f"] \arrow[d, "\langle o_1, t_1\rangle"'] & Q_2 \arrow[d, "\langle o_2, t_2\rangle"] \\ \mathbb{R} \times Q_1 \arrow[r, "1_{\mathbb{R}} \times f"'] & \mathbb{R} \times Q_2 \end{tikzcd}$$

(The function $1_{\mathbb{R}} \times f$ maps a pair $\langle r, q\rangle \in \mathbb{R} \times Q_1$ to $\langle r, f(q)\rangle \in \mathbb{R} \times Q_2$.)

The set $\mathbb{R}^\omega$ of all streams can itself be turned into a stream automaton. Defining $O : \mathbb{R}^\omega \to \mathbb{R}$ by $O(\sigma) = \sigma(0)$ and $T : \mathbb{R}^\omega \to \mathbb{R}^\omega$ 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 $l : Q \to \mathbb{R}^\omega$:

$$\begin{tikzcd} Q \arrow[r, dashed, "\exists ! l"] \arrow[d, "\langle o, t\rangle"'] & \mathbb{R}^\omega \arrow[d, "\langle O, T\rangle"] \\ \mathbb{R} \times Q \arrow[r, dashed, "1_{\mathbb{R}} \times l"'] & \mathbb{R} \times \mathbb{R}^\omega \end{tikzcd}$$

Proof. Let $(Q, \langle o, t\rangle)$ be an automaton and let the function $l : Q \to \mathbb{R}^\omega$ assign to a state q in Q the stream $(o(q), o(t(q)), o(t(t(q))), \ldots)$. 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 of Theorem 2.1. $\square$

3. Behavioural differential equations

In this section we define streams and operators on streams in stream calculus 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 Rutten (2000a). Here we shall summarise the main ideas by treating two typical examples, and refer the reader to Rutten (2000a) for more details. Moreover, the Appendix contains a theorem stating the unique existence of solutions for a rather general family of behavioural differential equations, comprising nearly 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:

$$\sigma' = \sigma, \quad \sigma(0) = 1.$$

This example is so simple that the solution is immediate: take $\sigma = (1, 1, 1, \ldots)$ 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 non-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 $t : S \to S$ by $t(s) = s$ and $o : S \to \mathbb{R}$ 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 $l : S \to \mathbb{R}^\omega$. We can now define $\sigma = l(s)$. Because l is a homomorphism, we do indeed have $\sigma' = T(\sigma) = T(l(s)) = l(t(s)) = l(s) = \sigma$, and $\sigma(0) = O(\sigma) = O(l(s)) = o(s) = 1$. 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 of Theorem 2.1, from the fact that $\{\langle \sigma, \rho \rangle\}$ is a bisimulation relation of $\mathbb{R}^\omega$. This shows that σ is the only solution of the differential equation.

So that 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 is usually 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 full 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:

$$(\sigma + \tau)' = \sigma' + \tau', \qquad (\sigma + \tau)(0) = \sigma(0) + \tau(0)$$

$$(\sigma \otimes \tau)' = (\sigma' \otimes \tau) + (\sigma \otimes \tau'), \qquad (\sigma \otimes \tau)(0) = \sigma(0) \times \tau(0).$$

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

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

and so on. This illustrates the computational 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-hand 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 $\mathscr{E}$ of expressions be given by the following syntax:

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

The set $\mathscr{E}$ 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 $\mathscr{E}$ by a syntactic counterpart, denoted by the same symbols $+$ and $\otimes$ again. The set $\mathscr{E}$ is next supplied with an automaton structure $(\mathscr{E}, \langle o, t \rangle)$ by defining functions $o : \mathscr{E} \to \mathbb{R}$ and $t : \mathscr{E} \to \mathscr{E}$. For the definition of o and t on the constant symbols $\underline{\sigma}$, we use the automaton structure $(\mathbb{R}^\omega, \langle O, T \rangle)$ on $\mathbb{R}^\omega$:

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

Thus the constant $\underline{\sigma}$ behaves in the automaton $\mathscr{E}$ precisely as the stream σ behaves in the automaton $\mathbb{R}^\omega$. (This includes $\mathbb{R}^\omega$ as a subautomaton in $\mathscr{E}$.) 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 $\mathscr{E}$, 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 $\mathscr{E}$ has now been turned into an automaton $(\mathscr{E}, \langle o, t \rangle)$, and because $\mathbb{R}^\omega$ is a final automaton, there exists, by Theorem 2.2, a unique homomorphism $l : \mathscr{E} \to \mathbb{R}^\omega$, which assigns to each expression E the stream $l(E)$ that 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 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 the Appendix, stating the well-definedness of a large class of (systems of) behavioural differential equations.

From now on we will take the existence of solutions of behavioural differential equations for granted, but 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 $\sigma = (1, 1, 1, \ldots)$ of the first example above can be expressed by means of the 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

In this section we introduce a number of basic stream operators by means of behavioural differential equations, and describe some elementary properties and examples. 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, 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:

$$[r]' = [0], \quad [r](0) = r.$$

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

$$[r] = (r, 0, 0, 0, \ldots).$$

Having once formally introduced the inclusion of the reals into the set of streams by means of this operator $[\ ] : \mathbb{R} \to \mathbb{R}^\omega$, we immediately observe that in stream calculus there will be hardly 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

$$X' = [1], \quad X(0) = 0.$$

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:

$$X = (0, 1, 0, 0, 0, \ldots).$$

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:

$$(\sigma + \tau)' = \sigma' + \tau', \quad (\sigma + \tau)(0) = \sigma(0) + \tau(0).$$

(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 $\sigma = (s_0, s_1, s_2, \ldots)$ and $\tau = (t_0, t_1, t_2, \ldots)$ is given by

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

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

$$\begin{aligned} \sigma + 0 &= \sigma \\ \sigma + \tau &= \tau + \sigma & (3) \\ \sigma + (\tau + \rho) &= (\sigma + \tau) + \rho. & (4) \end{aligned}$$

We can now go through 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

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

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

As motivation for the next operator, let us look briefly at the world of (real-valued) functions, which on more than one occasion will be a source of inspiration for stream calculus. Consider a function $f : \mathbb{R} \to \mathbb{R}$ and a stream (of coefficients) $\sigma = (s_0, s_1, s_2, \ldots)$ such that $f(x) = s_0 + s_1 x + s_2 x^2 + \cdots$. Writing $\bar{f}(x) = s_1 + s_2 x + s_3 x^2 + \cdots$, we have

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

Let $g : \mathbb{R} \to \mathbb{R}$ and $\tau = (t_0, t_1, t_2, \ldots)$ be another such function and stream, with $g(x) = t_0 + t_1 x + t_2 x^2 + \cdots$, and write, similarly, $g(x) = t_0\ +\ (x \times \bar{g}(x))$. Now computing 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 to motivate 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. We will say more about this later.) Let the (*convolution*) *product* $\sigma \times \tau$ of two streams σ and τ be the unique stream satisfying

$$(\sigma \times \tau)' = (\sigma' \times \tau) + (\sigma(0) \times \tau'), \quad (\sigma \times \tau)(0) = \sigma(0) \times \tau(0).$$

(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 $n \geqslant 0$,

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

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 is, indeed, *not* valid in general. 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 that 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 stream derivative and convolution product operations, 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 $n \geqslant 0$,

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

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

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

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 $\sigma \in \mathrm{IR}^{\omega}$: $\sigma = \sigma(0) + (X \times \sigma')$.

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 from 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 $R = \{\langle \sigma, \sigma \rangle \mid \sigma \in \mathrm{IR}^{\omega}\} \cup \{\langle \sigma,\, \sigma(0) + (X \times \sigma')\rangle \mid \sigma \in \mathrm{IR}^{\omega}\}$. 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 then follows by coinduction. □

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:

$$\sigma \times (\tau + \rho) \;=\; (\sigma \times \tau) + (\sigma \times \rho). \tag{6}$$

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

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

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

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

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 $R \subseteq \mathrm{IR}^{\omega} \times \mathrm{IR}^{\omega}$ is a *bisimulation-up-to* if, for all $\sigma, \tau \in \mathrm{IR}^{\omega}$, we have if $\sigma R \tau$, then:

(1) $\sigma(0) = \tau(0)$.

(2) There exist $n \geqslant 0$, $\alpha_0, \ldots, \alpha_n$, $\beta_0, \ldots, \beta_n \in \mathrm{IR}^{\omega}$, such that $\sigma' = \alpha_0 + \cdots + \alpha_n$ and $\tau' = \beta_0 + \cdots + \beta_n$, and, for all $0 \leqslant i \leqslant n$, either $\alpha_i = \beta_i$ or $\alpha_i \, R \, \beta_i$.

The latter condition will usually be denoted by

$$\alpha_0 + \cdots + \alpha_n \, (\Sigma R) \, \beta_0 + \cdots + \beta_n,$$

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

Theorem 4.2 (Coinduction-up-to). For all $\sigma, \tau \in \mathrm{IR}^\omega$,

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

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

(1) $R \subseteq \bar{R}$.

(2) $\{\langle \sigma, \sigma \rangle \mid \sigma \in \mathrm{IR}^\omega\} \subseteq \bar{R}$.

(3) For all $\alpha_0, \alpha_1, \beta_0, \beta_1 \in \mathrm{IR}^\omega$, 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 IR^ω. Since $R \subseteq \bar{R}$, the theorem follows by (ordinary) coinduction (Theorem 2.1). □

Proof of 6 – continued. Clearly the relation Q introduced above is a bisimulation-up-to. The identity now follows by coinduction-up-to. □

Similarly, one proves

$$(\tau + \rho) \times \sigma \ =\ (\tau \times \sigma) + (\rho \times \sigma). \tag{7}$$

Stream product is also associative:

$$\sigma \times (\tau \times \rho) \ =\ (\sigma \times \tau) \times \rho. \tag{8}$$

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

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

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

Stream product is commutative,

$$\sigma \times \tau \ =\ \tau \times \sigma, \tag{9}$$

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 $r \in \mathbb{R}$, $\sigma \in \mathbb{R}^\omega$,

$$r \times \sigma = \sigma \times r \tag{10}$$

$$(X \times \sigma)' = \sigma \tag{11}$$

$$\sigma \times X = X \times \sigma. \tag{12}$$

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

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

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

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

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

As 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. Therefore, let $\sigma_0, \sigma_1, \sigma_2, \ldots$ be a sequence of streams such that, for all $k \geqslant 0$, $\sigma_0(k) + \sigma_1(k) + \sigma_2(k) + \cdots < \infty$. Such a sequence is called *summable*. We define the *generalised sum* by

$$\left(\sum_{n=0}^{\infty} \sigma_n\right)' = \sum_{n=0}^{\infty} (\sigma_n'), \quad \left(\sum_{n=0}^{\infty} \sigma_n\right)(0) = \sum_{n=0}^{\infty} \sigma_n(0).$$

We shall often also write

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

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 $n \geqslant 0$,

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

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

$$s_0,\; s_1 \times X^1,\; s_2 \times X^2,\; \ldots$$

is summable, and satisfies

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

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

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

$$\pi = \; p_0 + p_1 X + \cdots + p_n X^n \; = (p_0, p_1, \ldots, p_n, 0, 0, 0, \ldots).$$

For instance,

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

are both 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 $[r] = (r, 0, 0, 0, \ldots)$ with $r \neq 0$, clearly have $[r]^{-1} = (r^{-1}, 0, 0, 0, \ldots) = [r^{-1}]$ as their inverse. Now considering an arbitrary stream σ with $\sigma(0) \neq 0$, and assuming for the moment that σ^{-1} indeed existed, it would have to satisfy

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

This implies

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

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 $(\sigma^{-1})(0)$: taking the initial value on both sides of the equation (recall that 1 stands for the stream $[1] = (1, 0, 0, 0, \ldots)$) gives

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

whence $(\sigma^{-1})(0) = \sigma(0)^{-1}$. 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 $\sigma \in \mathrm{IR}^{\omega}$ with $\sigma \neq 0$, we define the (*multiplicative*) *inverse* σ^{-1} as the unique stream satisfying the following equation:

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

Whenever we write 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.)

While our preference for using behavioural differential equations for the definition of the operators has so far been 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 $n \geqslant 1$. 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 $\sigma, \tau \in \mathrm{IR}^{\omega}$,

$$\sigma \times \sigma^{-1} = 1 \tag{14}$$

$$\sigma^{-1} \times \sigma = 1 \tag{15}$$

$$(\sigma^{-1})^{-1} = \sigma \tag{16}$$
$$(\sigma \times \tau)^{-1} = \tau^{-1} \times \sigma^{-1}. \tag{17}$$

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. 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:

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

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

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

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. □

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

$$\frac{1}{1-X} = 1 + X + X^2 + X^3 + \cdots \tag{18}$$

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

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

Here are some further identities:

$$\begin{aligned}
X \times (s_0, s_1, s_2, \ldots) &= (0, s_0, s_1, s_2, \ldots) && (19)\\
\frac{1}{1+X} &= 1 - X + X^2 - X^3 + \cdots && (20)\\
\frac{1}{1-X^2} &= 1 + X^2 + X^4 + \cdots && (21)\\
\frac{1}{(1-X)^2} &= 1 + 2X + 3X^2 + 4X^3 + \cdots && (22)\\
\frac{1}{1-rX} &= 1 + rX + r^2X^2 + r^3X^3 + \cdots && (23)\\
\frac{X}{1+X^2} &= X - X^3 + X^5 - X^7 + \cdots. && (24)
\end{aligned}$$

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

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

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

$$\left\{ \left\langle \frac{1}{(1-X)^2} + \frac{k}{1-X},\ (k+1) + (k+2)X + (k+3)X^2 + \cdots \right\rangle \mid k \geqslant 0 \right\}$$

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 $\tau(0) = 0$, we define the *composition* $\sigma \circ \tau$ as the unique stream satisfying the following behavioural differential equation:

$$(\sigma \circ \tau)' = \tau' \times (\sigma' \circ \tau), \quad (\sigma \circ \tau)(0) = \sigma(0).$$

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 $\sigma = (s_0, s_1, s_2, \ldots)$ and τ with $\tau(0) = 0$,

$$\sigma(\tau) = s_0 + (s_1 \times \tau^1) + (s_2 \times \tau^2) + \cdots. \tag{25}$$

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

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

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

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

Because

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

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

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

$$\begin{aligned}
\sigma(X) &= \sigma \\
\sigma(-X) &= s_0 - s_1X + s_2X^2 - s_3X^3 + \cdots \\
&= (s_0, -s_1, s_2, -s_3, \ldots) \\
\frac{1}{2}(\sigma(X) + \sigma(-X)) &= (s_0, 0, s_3, 0, \ldots) \\
\sigma(X^2) &= s_0 + s_1X^2 + s_2X^4 + s_3X^6 + \cdots.
\end{aligned} \tag{26}$$

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:

$$\sigma^{(n)} + r_{n-1}\sigma^{(n-1)} + \cdots + r_1\sigma^{(1)} + r_0\sigma = \tau, \quad \sigma^{(k)}(0) = c_k, \ \ 0 \leqslant k \leqslant n-1,$$

where $n \geqslant 1$, $\sigma^{(k)}$ denotes the k-th stream derivative of σ, $r_0, \ldots, r_{n-1}$ are real numbers (interpreted as constant streams $[r_k]$), $c_0, \ldots, c_{n-1}$ 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 $r_0, \ldots, r_{n-1}$, the initial values $c_0, \ldots, c_{n-1}$, 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 convey the main idea. Let us begin with a simple example of a homogeneous equation:

$$\sigma'' - \sigma' - \sigma = 0, \quad \sigma(0) = 0, \quad \sigma'(0) = 1.$$

In order to solve this, we multiply the left- and right-hand sides of the equation by 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, which is non-homogeneous this time:

$$\sigma' - \sigma = (1 - X)^{-1}, \quad \sigma(0) = 1$$

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:

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

Here are some further examples, which you may wish to use to test your stream-calculus abilities:

$$\sigma' - 5\sigma = -2(1-2X)^{-1}, \quad \sigma(0) = 3$$

$$\tau'' + r^2\tau = 1, \quad \tau(0) = 1, \quad \tau'(0) = 0$$

$$\rho'' - \rho' = 2 + 6X, \quad \rho(0) = 1, \quad \rho'(0) = 0.$$

The solutions of these equations are

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

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:

$$\sigma^{(n)} + (\rho_{n-1} \times \sigma^{(n-1)}) + \cdots + (\rho_1 \times \sigma^{(1)}) + (\rho_0 \times \sigma) = \tau, \quad \sigma^{(k)}(0) = c_k, \quad 0 \leqslant k \leqslant n-1$$

where now not only τ but also $\rho_0, \ldots, \rho_{n-1}$ 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 $\rho_0, \ldots, \rho_{n-1}$, the initial values $c_0, \ldots, c_{n-1}$, and the operators of plus, product and inverse.

The proof for arbitrary $n \geqslant 1$ is no more difficult (just more writing) than the proof for $n = 1$, which is presented next. Consider the equation

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

with initial value $\sigma(0)$ and 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:

$$s_{n+k} + r_{k-1}s_{n+k-1} + \cdots + r_1 s_{n+1} + r_0 s_n = t_n$$

with initial values $s_0, \ldots, s_{k-1}$ and where $k \geqslant 1$ (the order of the equation), $n \geqslant 0$, the coefficients $r_0, \ldots, r_{n-1}$ are real numbers (the coefficient for s_{n+k} has been taken identical to 1 for convenience), and $t_0, t_1, t_2, \ldots$ is an arbitrary sequence of real numbers. Defining

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

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 by the constant stream X^n:

$$s_{n+k}X^n + r_{k-1}s_{n+k-1}X^n + \cdots + r_1 s_{n+1}X^n + r_0 s_n X^n = t_n X^n.$$

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

$$\sum_{n=0}^{\infty} s_{n+k}X^n + r_{k-1}\sum_{n=0}^{\infty} s_{n+k-1}X^n + \cdots + r_1 \sum_{n=0}^{\infty} s_{n+1}X^n + r_0 \sum_{n=0}^{\infty} s_n X^n = \sum_{n=0}^{\infty} t_n X^n.$$

Since, for any $\mathrm{i} \geqslant 0$,

$$\sigma^{(\mathrm{i})} = s_{\mathrm{i}} + s_{\mathrm{i}+1}X^1 + s_{\mathrm{i}+2}X^2 + \cdots,$$

which is an immediate corollary of Theorem 4.3, we obtain

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

with initial values

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

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

We will now look at a few examples. Here is possibly the most famous difference equation of all, which defines the Fibonacci numbers:

$$s_{n+2} - s_{n+1} - s_n = 0, \quad s_0 = 0, \quad s_1 = 1.$$

The method above transforms it into the behavioural differential equation

$$\sigma'' - \sigma' - \sigma = 0, \quad \sigma(0) = 0, \quad \sigma'(0) = 1,$$

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 what the n-th element $\sigma(n)$ looks like, ideally as a function of n. The method of partial fractions, which is 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

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

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

$$s_{n+1} - s_n = 1, \quad s_0 = 1.$$

Our method transforms it into

$$\sigma' - \sigma = (1 - X)^{-1}, \quad \sigma(0) = 1$$

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

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

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

$$s_{n+1} - 5s_n = -2^{n+1}, \quad s_0 = 3.$$

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

$$\sigma' - 5\sigma = -2(1 - 2X)^{-1}, \quad \sigma(0) = 3.$$

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:

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

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 $\sigma(0) > 0$,

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

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 to that for $\sqrt{\sigma}'$:

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

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

(1) $\langle \tau, \sqrt{\sigma} \rangle \in R$.
(2) $\{\langle \rho, \rho \rangle \mid \rho \in \mathrm{IR}^\omega\} \subseteq R$.
(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. □

(Just as an entertaining aside, suggested to us by Alexandru Baltag, note that the defining equation for square root immediately gives rise to a representation as a continued fraction:

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

For more on continued fractions in stream calculus see Rutten (2001; 2003).)

Here are a few basic properties and examples of square roots:

$$\sqrt{\sigma} \times \sqrt{\tau} = \sqrt{\sigma \times \tau} \tag{30}$$

$$\sqrt{\frac{1}{\sigma}} = \frac{1}{\sqrt{\sigma}} \tag{31}$$

$$\sqrt{(1,1,1,\ldots)} = \left(1, \frac{1}{2}, \frac{3}{8}, \frac{5}{16}, \ldots\right) \tag{32}$$

$$\sqrt{(1,2,3,\ldots)} = (1,1,1,\ldots) \tag{33}$$

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

Proof of identities 30–34. The first identity is immediate by Theorem 7.1 and the observation that

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

For identity (31), we calculate as follows:

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

Example (33) follows from

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

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. □

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:

$$\gamma' = \gamma \times \gamma, \quad \gamma(0) = 1.$$

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

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

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:

$$\begin{aligned}
\gamma = 1 + (X \times \gamma^2) \ &\text{iff} \ (4X^2 \times \gamma^2) - (4X \times \gamma) + 4X = 0 \\
&\text{iff} \ (4X^2 \times \gamma^2) - (4X \times \gamma) + 1 = 1 - 4X \\
&\text{iff} \ (2X \times \gamma - 1)^2 = 1 - 4X \\
&\text{iff} \ (2X \times \gamma) - 1 = -\sqrt{1-4X}
\end{aligned}$$

[by Theorem 7.1; note that $((2X \times \gamma) - 1)(0) = -1$].

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) $X = (0, 1, 0, 0, 0, \ldots)$ has no meaning because inverse is only defined for streams with initial value different from 0. Fortunately, we can still get rid of the '$X\times$' simply by stream differentiation, since $(X \times \gamma)' = \gamma$. Thus

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

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 that 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 in 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:

$$(\sigma \otimes \tau)' = (\sigma' \otimes \tau) + (\sigma \otimes \tau') \qquad (\sigma \otimes \tau)(0) = \sigma(0) \times \tau(0)$$

$$(\sigma^{-\underline{1}})' = -\sigma' \otimes (\sigma^{-\underline{1}} \otimes \sigma^{-\underline{1}}) \qquad \sigma^{-\underline{1}}(0) = \sigma(0)^{-1}.$$

(The shuffle inverse is defined only when $\sigma(0) \neq 0$.) 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 (we will say more about this later). Yet another way of explaining the relevance of shuffle product is provided by the following property. For streams $\sigma = (s_0, s_1, s_2, \ldots)$ and $\tau = (t_0, t_1, t_2, \ldots)$, we have

$$\left(s_0 + \frac{s_1}{1!}X^1 + \frac{s_2}{2!}X^2 + \cdots\right) \times \left(t_0 + \frac{t_1}{1!}X^1 + \frac{t_2}{2!}X^2 + \cdots\right)$$
$$= (\sigma \otimes \tau)(0) + \frac{(\sigma \otimes \tau)(1)}{1!}X^1 + \frac{(\sigma \otimes \tau)(2)}{2!}X^2 + \cdots, \qquad (36)$$

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

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

As in 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 $n \geqslant 0$,

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

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

$$\begin{aligned}
\sigma \otimes \tau &= \tau \otimes \sigma && (38)\\
r \otimes \sigma &= r \times \sigma && (39)\\
0 \otimes \sigma &= 0 && (40)\\
1 \otimes \sigma &= \sigma && (41)\\
\sigma \otimes (\tau \otimes \rho) &= (\sigma \otimes \tau) \otimes \rho && (42)\\
\sigma \otimes (\tau + \rho) &= (\sigma \otimes \tau) + (\sigma \otimes \rho) && (43)\\
(\sigma^{\underline{n+1}})' &= (n+1) \otimes \sigma' \otimes \sigma^{\underline{n}} && (44)\\
X^{\underline{n}} &= n! \times X^n && (45)\\
\sigma \otimes \sigma^{-\underline{1}} &= 1 && (46)\\
(\sigma^{-\underline{1}})^{-\underline{1}} &= \sigma && (47)\\
(\sigma \otimes \tau)^{-\underline{1}} &= \sigma^{-\underline{1}} \otimes \tau^{-\underline{1}} && (48)\\
X \otimes \sigma &= s_0X^1 + 2s_1X^2 + 3s_2X^3 + \cdots && (49)\\
&= (0, s_0, 2s_1, 3s_2, \ldots) &&\\
(X \otimes \sigma')' &= s_1 + 2s_2X^1 + 3s_3X^2 + \cdots && (50)\\
&= (s_1, 2s_2, 3s_3, \ldots) &&\\
(1 - X)^{-\underline{1}} &= 1 + 1!X + 2!X^2 + 3!X^3 \cdots. && (51)
\end{aligned}$$

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 proved with induction on n. For (46) define

$$Q = \{\langle \rho \otimes (\sigma \otimes \sigma^{-\underline{1}}),\, \rho\rangle \mid \rho, \sigma \in \mathrm{IR}^{\omega}\}.$$

Computing derivatives gives

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

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

$$T = \{\langle X \otimes \sigma,\, s_0X^1 + 2s_1X^2 + 3s_2X^3 + \cdots\rangle \mid \sigma = (s_0, s_1, s_2, \ldots) \in \mathrm{IR}^{\omega}\}$$

and note that

$$\begin{aligned}(X \otimes \sigma)' &= \sigma + (X \otimes \sigma') \\ &\;\Sigma T\; \sigma + (s_1X^1 + 2s_2X^2 + 3s_3X^3 + \cdots) \\ &= (s_0 + s_1X^1 + s_2X^2 + \cdots) + (s_1X^1 + 2s_2X^2 + 3s_3X^3 + \cdots) \\ &\qquad\qquad [\text{by Theorem 4.3}] \\ &= s_0 + 2s_1X^1 + 3s_2X^2 + 4s_3X^3 + \cdots \\ &= (s_0X + 2s_1X^2 + 3s_2X^3 + 4s_3X^4 + \cdots)',\end{aligned}$$

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

$$\{\langle n!(1-X)^{-\underline{n+1}},\, n! + (n+1)!X + (n+2)!X^2 + \cdots\rangle \mid n \geqslant 0\}$$

is a bisimulation relation on IR^{ω}. □

There are obvious variations for shuffle product and inverse for 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 $\sigma = (s_0, s_1, s_2, \ldots)$,

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

And the following behavioural differential equation defines an operation of 'shuffle square root',

$$\left(\sigma^{\underline{1/2}}\right)' = 1/2 \otimes \sigma' \otimes \left(\sigma^{\underline{1/2}}\right)^{-\underline{1}}, \quad \sigma^{\underline{1/2}}(0) = \sqrt{\sigma(0)},$$

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

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:

$$exp(\sigma)' = \sigma' \otimes exp(\sigma), \quad exp(\sigma)(0) = e^{\sigma(0)}$$

where $e^{\sigma(0)}$ is the analytical function e^x applied to the real number $\sigma(0)$. Exponentiation has many familiar properties, which we shall formulate shortly. However, the most important property for our purposes is less familiar, and, 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:

$$\begin{aligned}
exp(\sigma) \;&=\; 1 + \frac{\sigma^{\underline{1}}}{1!} + \frac{\sigma^{\underline{2}}}{2!} + \cdots && (53)\\
exp(\sigma) \otimes exp(\tau) \;&=\; exp(\sigma + \tau) && (54)\\
exp(-\sigma) \;&=\; exp(\sigma)^{-\underline{1}}\,. && (55)
\end{aligned}$$

The following related identities will come in handy later:

$$\begin{aligned}
\frac{1}{1-rX} \otimes \frac{1}{1-sX} \;&=\; \frac{1}{1-(r+s)X} && (56)\\
\left(\frac{1}{1-rX}\right)^{\underline{n}} \;&=\; \frac{1}{1-nrX} && (57)\\
((1+rX)^n)' \;&=\; r(1+rX)^{n-1} + r(1+rX)^{n-2} + \cdots + r(1+rX) + r && (58)\\
\left(\frac{1}{(1-rX)^n}\right)' \;&=\; \frac{r}{(1-rX)^n} + \frac{r}{(1-rX)^{n-1}} + \cdots + \frac{r}{1-rX} && (59)\\
X^{n+1} \times \frac{1}{1-X} \;&=\; -1 - X - \cdots - X^n + \frac{1}{1-X}. && (60)
\end{aligned}$$

Proof of identities 52–60. Again we will 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 $Q = \{\langle \tau \otimes exp(-\sigma),\, \tau \otimes exp(\sigma)^{-\underline{1}}\rangle \mid \sigma, \tau \in \mathbb{R}^\omega\}$ and compute as follows:

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

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

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

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:

$$s_{n+1} - (n+1)s_n = 0, \quad s_0 = 1.$$

It only takes a moment's thought to see that the stream $(0!, 1!, 2!, \ldots)$ 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 soon, the two new operators of shuffle product and shuffle inverse will allow us to do so.

The stream $(0!, 1!, 2!, \ldots)$ is the prototypical example of a divergent stream, in the sense that if one were to define a function $f(x) = 0! + 1!x + 2!x + \cdots$, 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 Wilf (1994)), 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.* Aczel (1988, pages 346–348)).

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

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

As before, putting $\sigma = (s_0, s_1, s_2, \ldots)$, 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,

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

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

10. Comparing convolution product and shuffle product

In this section we investigate the relation between the two types of product and inverse. 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

$$\frac{d}{dX}(\sigma) \; = \; (X \otimes \sigma')'. \tag{61}$$

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,

$$\frac{d}{dX}(s_0 + s_1 X^1 + s_2 X^2 + \cdots) \; = \; s_1 + 2s_2 X^1 + 3s_3 X^2 + \cdots, \tag{62}$$

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 familiar way from analysis. In particular, there are the following identities:

$$\frac{d\,(\sigma \times \tau)}{dX} \; = \; \left(\frac{d\,\sigma}{dX} \times \tau\right) + \left(\sigma \times \frac{d\,\tau}{dX}\right) \tag{63}$$

$$\frac{d\,\sigma^{-1}}{dX} \; = \; -\frac{d\,\sigma}{dX} \times \sigma^{-1} \times \sigma^{-1} \tag{64}$$

$$\frac{d\,\sigma^{n+1}}{dX} \; = \; (n+1) \times \frac{d\,\sigma}{dX} \times \sigma^{n}. \tag{65}$$

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 $n \geqslant 0$,

$$\frac{d\left(\sum_{n=0}^{\infty} \sigma_n\right)}{dX} \; = \; \sum_{n=0}^{\infty} \frac{d\,\sigma_n}{dX}$$

$$\frac{d\,(X^n \times \tau)}{dX} \; = \; \left(\frac{d\,X^n}{dX} \times \tau\right) \; + \; \left(X^n \times \frac{d\,\tau}{dX}\right)$$

$$\frac{d\,X^{n+1}}{dX} \; = \; (n+1) \times X^n.$$

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

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

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

Identities (64) and (65) follow easily from (63). □

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

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

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

$$
\Lambda_c(\sigma)' = \Lambda_c \left(\frac{d\,\sigma}{dX} \right), \quad \Lambda_c(\sigma)(0) = \sigma(0).
$$

One can easily prove that Λ_c transforms a stream $(s_0, s_1, s_2, s_3, \ldots)$ into

$$
(0!s_0, 1!s_1, 2!s_2, 3!s_3, \ldots),
$$

or, equivalently,

$$
\Lambda_c \left(s_0 + \frac{s_1}{1!} X + \frac{s_2}{2!} X^2 + \frac{s_3}{3!} X^3 + \cdots \right) = s_0 + s_1 X + s_2 X^2 + s_3 X^3 + \cdots.
$$

In combinatorics, this is referred to as the *Laplace–Carson* transform (*cf.* Bergeron *et al.* (1998, page 350) and Comtet (1974 page 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 $r \in \mathbb{R}$, $\sigma, \tau \in \mathbb{R}^\omega$,

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

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) $\langle \Lambda_c((\sigma_1)^{-1}),\ (\tau_1)^{-\underline{1}} \rangle \in R$.

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:

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

We should emphasise, 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 $\sigma \in \mathbb{R}^\omega$:

$$1 = \frac{1}{1+rX} \otimes \frac{1}{1-rX} \tag{66}$$

$$X \otimes \sigma = \left(X^2 \times \frac{d\,\sigma}{dX} \right) + (X \times \sigma) \tag{67}$$

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

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

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

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

And for (68), define

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

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

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

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

To illustrate identity (67), we compute

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

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):

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

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

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 $\sigma = (s_0, s_1, s_2, \ldots)$ be defined as

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

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

$$\sigma \otimes \frac{1}{1+X} = (\Delta^0\sigma)(0) + (\Delta^1\sigma)(0) \times X^1 + (\Delta^2\sigma)(0) \times X^2 + \cdots. \tag{73}$$

Proof of 73. First note that for all $n \geqslant 0$,

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

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

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

Identity (73) now follows by coinduction-up-to. □

Equivalently, we have

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

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

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

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

Theorem 11.1. For any stream σ,

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

Proof.

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

□

This derivation may be compared with the one of, for instance, Scheid (1968, 11.38), where the same formula can be derived only under a number of convergence assumptions.

12. Application: solving analytical differential equations

The method of undetermined coefficients (*cf.* Birkhoff and Rota (1978, page 82)) is a classical technique in analysis for the solution of differential equations defining analytical functions. The idea is quickly explained by means of an example. In order to solve the differential equation

$$f'' + f = 0, \quad f(0) = 0, \quad f'(0) = 1,$$

one assumes the solution to be of the shape

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

Computing f'' gives

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

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

$$s_{n+2} + s_n = 0, \quad s_0 = 0, \quad s_1 = 1.$$

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. First, more interesting differential equations quickly lead to very complicated difference equations. Second, 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 above method, which in many applications is free from these restrictions. Defining

$$\mathscr{A} = \{ f : \mathrm{IR} \to \mathrm{IR} \mid f \text{ is analytic in (a neighbourhood of) } 0 \ \},$$

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

$$\mathscr{T}(f)' = \mathscr{T}(f'), \quad \mathscr{T}(f)(0) = f(0).$$

(Note that the first occurrence of $'$ in $\mathscr{T}(f)' = \mathscr{T}(f')$ stands for stream derivation, whereas the second denotes analytical function derivation.) This definition is a variation on a definition by Pavlović and Escardó (Pavlović and Escardó 1998), who characterised $\mathscr{T}$ as a final coalgebra homomorphism in order to give a coinductive characterisation of the Laplace transform. Here $\mathscr{T}$ is studied in its own right, and will serve rather as an alternative to a Laplace transform. For that reason, we shall sometimes refer to the stream $\mathscr{T}(f)$ as the *Taylor transform* of f.

Our method is characterised by the following three steps:

(1) The function $\mathscr{T}$ is used to transform, in a systematic fashion, a differential equation for f into a behavioural differential equation for the stream $\mathscr{T}(f)$ 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, the following theorem expresses that the function $\mathscr{T}$ does indeed transform functions into their Taylor series in a systematic manner. For functions f and g, we shall be using the following familiar definitions: for all $x \in \mathbb{R}$,

$$\begin{aligned} (f+g)(x) &= f(x) + g(x) \\ (f \cdot g)(x) &= f(x) \times g(x) \\ f^{-1}(x) &= f(x)^{-1} \\ e^f(x) &= e^{f(x)}. \end{aligned}$$

Theorem 12.1. For all analytic functions $f, g \in \mathscr{A}$,

$$\mathscr{T}(f)(0) = f(0) \tag{74}$$
$$\mathscr{T}(f') = \mathscr{T}(f)' \tag{75}$$
$$\mathscr{T}(f+g) = \mathscr{T}(f) + \mathscr{T}(g) \tag{76}$$
$$\mathscr{T}(f \cdot g) = \mathscr{T}(f) \otimes \mathscr{T}(g) \tag{77}$$
$$\mathscr{T}(f^{-1}) = \mathscr{T}(f)^{-\underline{1}} \tag{78}$$
$$\mathscr{T}(e^f) = exp(\mathscr{T}(f)). \tag{79}$$

The most useful instantiation of identity (79) will be the case $f(x) = rx$: see indentity (82) below. Identities (74)–(77) are implicitly present in Pavlović and Escardó (1998); see also Rutten (2000a).

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

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

and similarly for the other elements of R. Now the theorem follows by coinduction. □

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 $r \in \mathbb{R}$, $n \geqslant 0$, $f \in \mathcal{A}$,

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

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

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

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

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

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

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

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

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

and use coinduction. For (85), note that

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

Finally, for (86), we have:

$$\begin{aligned}
\mathscr{T}(f \cdot x) &= \mathscr{T}(f) \otimes \mathscr{T}(x) && \text{[identity (77)]} \\
&= \mathscr{T}(f) \otimes X && \text{[identity (81)]} \\
&= \left(X^2 \times \frac{d\,\mathscr{T}(f)}{dX}\right) + (X \times \mathscr{T}(f)) && \text{[identity (67)].}
\end{aligned}$$

□

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

(1) To warm up, consider the equation from the beginning of the present section:

$$f'' + f = 0, \quad f(0) = 0, \quad f'(0) = 1.$$

Applying $\mathscr{T}$ to both sides of the equation and putting $\sigma = \mathscr{T}(f)$ gives the following behavioural differential equation:

$$\sigma'' + \sigma = 0, \quad \sigma(0) = 0, \quad \sigma'(0) = 1.$$

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

$$\sigma = X - X^3 + X^5 - X^7 + \cdots,$$

which, in combination with identity (81) gives

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

We can often 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 with an answer immediately:

$$f(x) = sin(x).$$

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

$$f' - f = e^x, \quad f(0) = 1.$$

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

$$\sigma' - \sigma = (1 - X)^{-1}, \quad \sigma(0) = 1.$$

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

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

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

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

Now using identities (76), (77), and (82), we get

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

(3) For a third example consider

$$f'' + r^2 f = 1, \quad f(0) = 1, \quad f'(0) = 1.$$

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

$$\sigma'' + r^2\sigma = 1, \quad \sigma(0) = 1, \quad \sigma'(0) = 1.$$

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

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

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) For a fourth example consider

$$f'' - f' = 2 + 6x, \quad f(0) = 1, \quad f'(0) = 0.$$

Step (1) gives

$$\sigma'' - \sigma' = 2 + 6X, \quad \sigma(0) = 1, \quad \sigma'(0) = 0.$$

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

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

Applying these, 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 we introduce the notion of a weighted stream automaton (called a non-deterministic automaton in Rutten (2000a)), the transition diagrams of which can serve as pleasant graphical representations of streams and their successive derivatives. Finite weighted stream automata turn out to correspond to rational streams, but as we shall see, infinite weighted automata can be useful too.

A IR-*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 $o : Q \to \mathrm{IR}$, and a transition function $t : Q \to (Q \to_f \mathrm{IR})$, where the latter set only contains functions of finite *support*:

$$Q \to_f \mathrm{IR} = \{\phi : Q \to \mathrm{IR} \mid |\{q \in Q \mid \phi(q) \neq 0\}| < \infty\}.$$

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

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

$$q \overset{r}{\Rightarrow} \text{ iff } o(q) = r,$$

which will allow us to present weighted automata using pictures. In these pictures, we will only draw those arrows that have a non-zero label. For $q \in Q$, let $\{q_1, \ldots, q_n\}$ 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 $1 \leqslant i \leqslant n$. The following diagram contains all the relevant information about the state q in the automaton Q:

$$\begin{tikzcd} & q \arrow[r, Rightarrow, "o(q)"] \arrow[dl, "r_1"'] \arrow[dr, "r_n"] & {} \\ q_1 & \cdots & q_n \end{tikzcd}$$

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. We shall also use the following convention. Labels that are 1 are often omitted, and if $o(q) = 1$, we will call q an *output state*, which will often be denoted by underlining q:

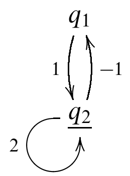
$$q \longrightarrow q' \;\equiv\; q \xrightarrow{1} q', \quad \underline{q} \equiv q \overset{1}{\Longrightarrow}.$$

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

$$S(q)' = r_1 S(q_1) + \cdots + r_n S(q_n), \quad S(q)(0) = o(q).$$

(As before, $\{q_1, \ldots, q_n\}$ is the support of $t(q)$ and $r_i = t(q)(q_i)$, for $1 \leqslant i \leqslant n$.) The pair (Q, q) is called a *representation* of the stream $S(q)$. A stream $\sigma \in \mathrm{IR}^\omega$ 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 *recognisable*.

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



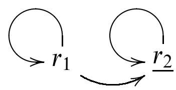
We have the following equations for the behaviour of q_1 and q_2:

$$S(q_1)' = S(q_2), \quad S(q_1)(0) = 0$$
$$S(q_2)' = -S(q_1) + 2S(q_2), \quad S(q_2)(0) = 1.$$

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:



(where all labels are 1 and thus have been omitted, and where r_2 has output value 1 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. This proposition provides some operational intuition about the behaviour of weighted automata (but it is not very suited for reasoning about them).

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

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

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

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

$Q = \{q_1, \ldots, q_n\}$ and let μ be the $n \times n$ matrix with entries $\mu_{ij} = t(q_i)(q_j)$. Furthermore, write $o : Q \to \mathbb{R}$ as a column vector $o^t = (o(q_1), \ldots, o(q_n))^t$.

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

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

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 $n \geqslant 1$, real numbers s_{ij}, for $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant n$, real numbers $k_1, \ldots, k_n$, and streams $\sigma_1, \sigma_2, \ldots, \sigma_n$ with $\sigma = \sigma_1$, satisfying the following finite system of behavioural differential equations:

$$(\sigma_1)' = s_{11}\sigma_1 + s_{12}\sigma_2 + \cdots + s_{1n}\sigma_n, \quad \sigma_1(0) = k_1$$

$$(\sigma_2)' = s_{21}\sigma_1 + s_{22}\sigma_2 + \cdots + s_{2n}\sigma_n, \quad \sigma_2(0) = k_2,$$

$$\ldots$$

$$(\sigma_n)' = s_{n1}\sigma_1 + s_{n2}\sigma_2 + \cdots + s_{nn}\sigma_n, \quad \sigma_n(0) = k_n.$$

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

Proof. A complete proof can be found in Rutten (2000a). (See also Section 15 for further references on rationality.) Most interesting is the proof that (1) implies (2), which is as follows. Consider polynomials $\pi = p_0 + p_1X + \cdots + p_nX^n$ and $\rho = r_0 + r_1X + \cdots + r_mX^m$ 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 $m \leqslant n$ can be dealt with similarly). Using the defining behavioural differential equations of sum, convolution product, and inverse, σ is easily seen to satisfy the following finite system of behavioural differential equations:

$$(\pi \times \rho^{-1})' = v_1\rho^{-1} + v_2X\rho^{-1} + \cdots + v_mX^{m-1}\rho^{-1}, \quad (\pi \times \rho^{-1})(0) = p_0$$

$$(\rho^{-1})' = -r_1\rho^{-1} - r_2X\rho^{-1} - \cdots - r_mX^{m-1}\rho^{-1}, \quad (\rho^{-1})(0) = 1$$

$$(X\rho^{-1})' = \rho^{-1}, \quad (X\rho^{-1})(0) = 0$$

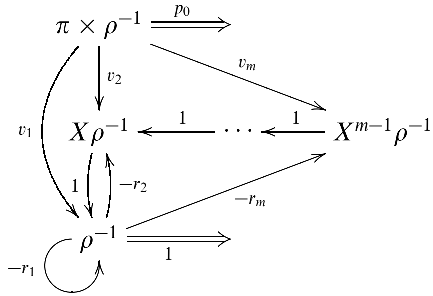
$$\ldots$$

$$(X^{m-1}\rho^{-1})' = X^{m-2}\rho^{-1}, \quad (X^{m-1}\rho^{-1})(0) = 0$$

where

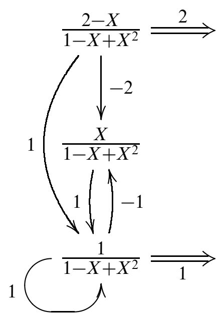
$$v_i = \begin{cases} p_i - p_0 r_i & \text{if } 1 \leqslant i \leqslant n \\ -p_0 r_i & \text{if } n < i \leqslant m. \end{cases}$$

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 ($\pi \times \rho^{-1}$, $X\rho^{-1}$, and so on) 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 also apply this 'splitting derivatives' procedure in the construction of infinite weighted automata for non-rational streams.

Recall that in Theorem 10.2, a number of identities for shuffle elimination were 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

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

and

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

that 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 with initial values $\{(\sigma_i \otimes \tau_j)(0) = k_i \times l_j\}_{i,j}$:

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

Thus $\sigma \otimes \tau$ is rational, again by Theorem 13.3. □

The proof of Theorem 13.4 in fact describes a general procedure 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 $\sigma \otimes \tau$ with the obvious initial values:

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

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

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

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, which 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 that it is also worth studying infinite weighted automata representing non-rational streams. We hope the next example will convince you.

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

$$tan' = 1 + tan^2, \quad tan(0) = 0.$$

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

$$\tau' = 1 + (\tau \otimes \tau), \quad \tau(0) = 0.$$

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 the ‘splitting of derivatives’ procedure again 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 $\tau \otimes \tau = \tau^{\underline{2}}$. Computing the derivative of the latter, we find, using the differential equation for τ again,

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

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

$$\begin{tikzcd} \underline{1} & \tau \arrow[l, "1"'] \arrow[r, bend left, "1"] & \tau^{\underline{2}} \arrow[l, bend left, "2"] \arrow[r, bend left, "2"] & \tau^{\underline{3}} \arrow[l, bend left, "3"] \arrow[r, bend left, "3"] & \tau^{\underline{4}} \arrow[l, bend left, "4"] \arrow[r, bend left, "4"] & \cdots . \arrow[l, bend left, "5"] \end{tikzcd}$$

(All output values are 0 except that of the state 1, which is 1.) 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:

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

A proof by coinduction is easily given, but see Rutten (2003), where far more general results are proved.

We conclude the present section with a coalgebraic characterisation 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 all:

$$\begin{tikzcd} Q \arrow[r, "\{\cdot\}"'] \arrow[d, "\langle o,t\rangle"'] \arrow[rr, bend left, "S"] & (Q \to_f \mathbb{R}) \arrow[r, dashed, "\exists! l"'] \arrow[dl, "\langle \hat{o},\hat{t}\rangle"] & \mathbb{R}^\omega \arrow[d, "\langle O,T\rangle"] \\ \mathbb{R} \times (Q \to_f \mathbb{R}) \arrow[rr, dashed, "1_{\mathbb{R}} \times l"'] & & \mathbb{R} \times \mathbb{R}^\omega \end{tikzcd}$$

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

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

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

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

With these functions, we have provided $(Q \to_f \mathbb{R})$ with an ordinary (that is, as in Section 2) stream automaton structure $((Q \to_f \mathbb{R}), \langle \hat{o}, \hat{t} \rangle)$. 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 non-deterministic 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 $l : (Q \to_f \mathbb{R}) \to \mathbb{R}^\omega$. 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. Application: coinductive counting

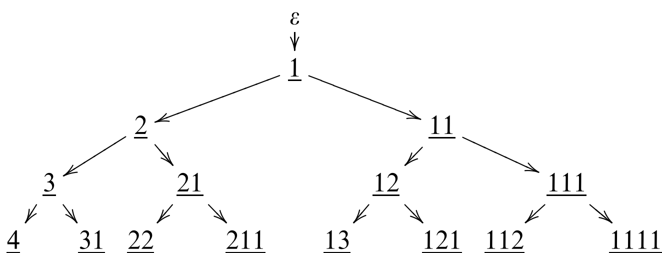
Weighted automata can be used to tackle, in a uniform and fairly general way, many so-called counting problems stemming from the world of enumerative combinatorics (*cf.* Aczel (1988), Flajolet and Sedgewick (1993; 2001) and Stanley (1997; 1999)). The resulting methodology, which we have called coinductive counting, consists of three steps:

(1) *Enumerate* the objects to be counted in an infinite, tree-shaped weighted automaton.
(2) *Identify* states that represent identical streams, using bisimulation.
(3) *Express* the resulting stream of counts in terms of stream constants and operators.

We present a few examples taken from Rutten (2003), which contains many more.

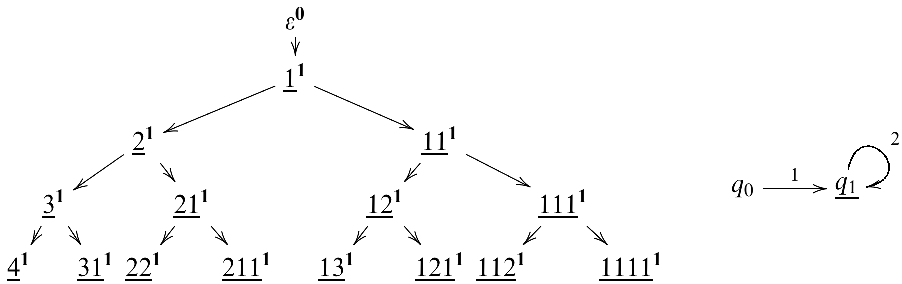
A *composition* of a natural number $k \geqslant 0$ is a sequence of natural numbers $n_1 \cdots n_l$ such that $k = n_1 + \cdots + n_l$. We now ask what is, for any $k \geqslant 0$, the number s_k of compositions of k? We present the answer by performing the three steps mentioned above:

(1) The following weighted automaton enumerates all compositions for all natural numbers (here and in what follows, pictures show only the first few levels of what is understood to be an infinite automaton):



Note that we have 1-transitions only (the labels are omitted) and that all states (except the first) have output value 1. The k-th level of this automaton contains all compositions of the natural number k. It is an immediate consequence of Proposition 13.1, therefore, that the initial state ε represents the stream $\sigma = (s_0, s_1, s_2, \ldots)$ of answers we are after: $S(\varepsilon) = \sigma$.

(2) Next we identify as many states as possible by defining a bisimulation-up-to between (the streams represented by) our weighted automaton, repeated below on the left, and the tiny 2 state automaton on the right:



The superscripts that we have added to the states of our automaton on the left, indicate to which state in the automaton on the right they are related. Or, more explicitly, the above picture suggests the definition of a relation $R \subseteq \mathbb{R}^\omega \times \mathbb{R}^\omega$ as

$$R = \{\langle S(\varepsilon), (q_0)\rangle\} \cup \{\langle S(w), S(q_1)\rangle \mid w \in \mathbb{N}^*,\ w \neq \varepsilon\}.$$

It is easy to check that R is indeed a bisimulation-up-to: all initial values match; $S(\varepsilon)' = S(1)$, which is related to $S(q_1) = S(q_0)'$; and for all words $v \in \mathbb{N}^*$ and natural numbers n, writing vn for the concatenation of v and n, we have $S(vn)' = S(v(n+1)) + S(vn1)$, each component of which is related to $S(q_1)$, thus matching $S(q_1)' = 2 \times S(q_1) = S(q_1) + S(q_1)$. It follows by coinduction-up-to that $\sigma = S(\varepsilon) = S(q_0)$.

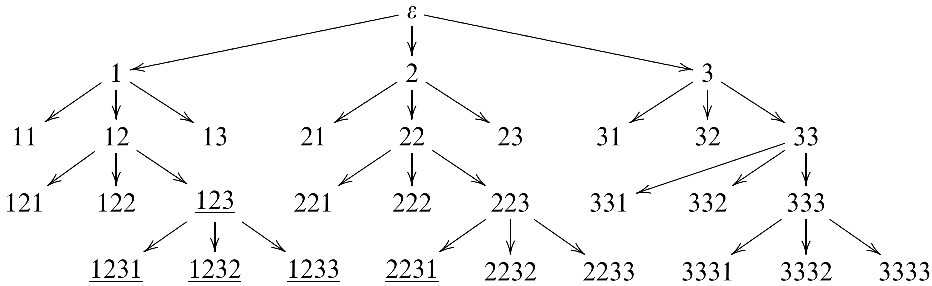
(3) The latter can be easily computed:

$$S(q_0) = \frac{X}{1 - 2X} \quad (= (0, 2^0, 2^1, 2^2, \ldots)).$$

It is worth emphasising the quantitative aspect of the notion of bisimulation (up-to): the fact that any state of the original weighted automaton labelled by a non-empty word w can take *two* transitions to similar such states is reflected by a 2-*step* from q_1 to itself.

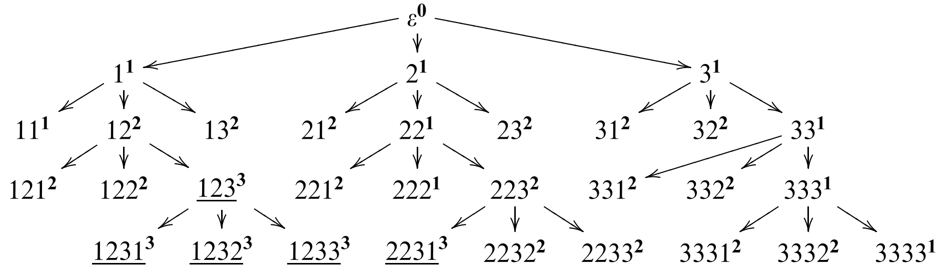
As a second example, we ask what is, for any natural number $k \geqslant 0$, the number s_k of surjections from the set $\{1, \ldots, k\}$ onto the set $\{1, 2, 3\}$ (defining s_0 to be 0)? Below we shall see how the answer can be generalised to surjections onto the set $\{1, \ldots, n\}$, for a fixed but arbitrary $n \geqslant 1$.

(1) Let us denote a function $f : \{1,\ldots,k\} \to \{1,2,3\}$ by means of the word $f(1)\cdots f(k)$. The following automaton enumerates at each level k all such functions:

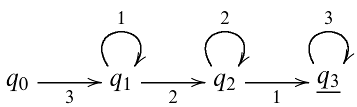


Note that all states labelled by a word representing a surjection (that is, containing at least one 1, one 2, and one 3), have been defined as output states. Also note that we have not only restricted the picture to the first five levels, but that, moreover, not all transitions have been included, for lack of space. As before, it follows from Proposition 13.1 that the initial state ε represents the stream $\sigma = (s_0, s_1, s_2, \ldots)$ of answers we are interested in.

(2) The automaton can be simplified by identifying all states (labelled with a word) containing an equal number of different symbols, as indicated by the superscripts below:



If one relates (the streams represented by) all i-superscripted states above with the state q_{i} in the automaton



one obtains a bisimulation-up-to, from which $S(\varepsilon) = S(q_0)$ follows by coinduction-up-to.

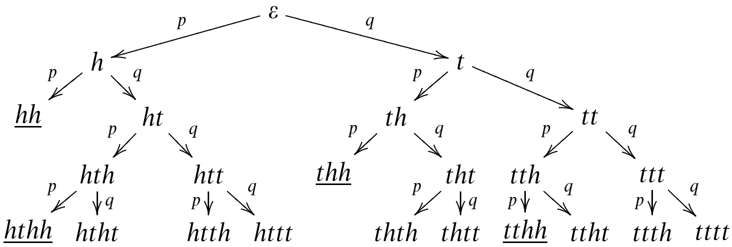
(3) The latter stream can be easily computed, yielding

$$\sigma = S(q_0) \;=\; \frac{3!X^3}{(1-X)(1-2X)(1-3X)}.$$

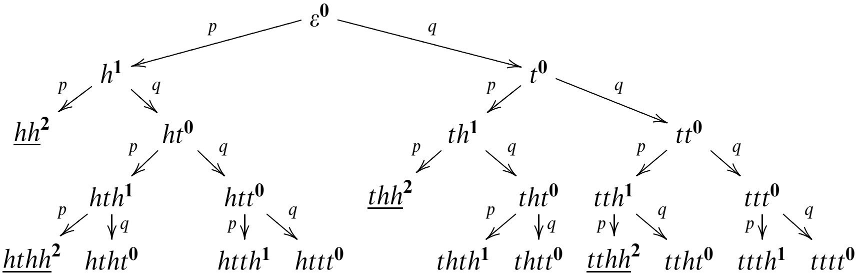
The formula for surjections onto the set $\{1,\ldots,n\}$, for arbitrary $n \geqslant 1$, can also be found without much more work: $n!X^n/(1-X)(1-2X)\cdots(1-nX)$.

For a final example, consider a, not necessarily fair, coin with probability p of producing a head and probability $q = 1 - p$ of producing a tail. We ask what is, for any $k \geqslant 0$, the probability s_k of getting, by flipping the coin k times, a sequence of heads and tails (of length k) without the occurrence of two consecutive heads apart from the two very last outcomes, which are required to be heads?

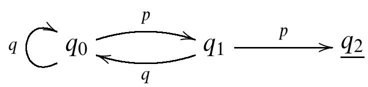
(1) Here is a weighted automaton describing all possible scenarios:



(2) All states that are (labelled with a sequence) ending in two heads are output states, and have no further transitions. States can be identified according to the number of final heads:



yielding the automaton



(3) The corresponding formula for $\sigma = (s_0, s_1, s_2, \ldots)$ is

$$\sigma = S(q_0) = \frac{p^2 X^2}{1 - qX - pqX^2} \,.$$

15. Discussion and related work

General references on (universal) coalgebra are Jacobs and Rutten (1997) and Rutten (2000b). Earlier work on a coalgebraic approach to automata, formal languages, and formal power series include Rutten (1998; 1999). The formulation of a coinductive proof principle in terms of bisimulation relations goes back to work by Aczel and Mendler (Aczel 1988; Aczel and Mendler 1989), which generalises Park and Milner's notion of bisimulation (Park 1981; Milner 1980) to a categorical setting. The notion of stream derivative is a special instance of the notion of *input derivative*, which goes back to Brzozowski (1964).

It also plays a role in Conway (1971), 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 (Milner 1989) (see also Sangiorgi (1998)). In Bartels (2000), 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 Bartels (2000), to a version that would allow derivatives to be bisimilar up to arbitrary contexts (including product, inverse, and the other operators). See also Lenisa (1998; 1999). The proofs of, for instance, identities (15) and (25) could be simplified if we had used bisimulation-up-to product.

The present paper is a reworking of Rutten (2001). It extends Rutten (2000a), 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. Because the present paper is already long enough as it is, we have not dealt with formal power series in many non-commutative variables, sometimes called multivariate streams, which were treated in Rutten (2000a) and Rutten (2001).

Pavlović and Escardó's paper (Pavlović and Escardó 1998) on calculus in coinductive form, which emphasises 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. However, apart from 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 Aczel (1988) and Wilf (1994), and the papers McIlroy (1999; 2001) and Karczmarczuk (1997; 2000).

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 Aczel (1988) and Wilf (1994)), functions (from $\mathbb{R}$ to $\mathbb{R}$) are just not needed in stream calculus, so convergence issues 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 also see some advantages of stream calculus over the use of formal power series. First, there is the rigorous use of coinduction, both in definitions and in proofs, which may make stream calculus 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. Second, stream calculus is a more expressive calculus, because of the simultaneous presence of:

— two types of multiplication (convolution product and shuffle product);

— the corresponding two types of inverse;

— two types of derivatives (ordinary and analytical).

This is, for instance, illustrated by the example of a divergent recurrence in Section 9. Moreover, the interplay between the various operators, notably between convolution inverse and shuffle product 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 Scheid (1968, 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.* Scheid (1968, page 75)).

Solving differential equations in stream calculus essentially amounts to the classical method of undetermined coefficients (Birkhoff and Rota 1978, page 82), with the difference that the difference equations obtained 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.* Sneddon (1972) and Mikusinski (1983)), because the operation of assigning the Taylor series $\mathscr{T}(f)$ 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.* Sneddon (1972, 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).

See Berstel and Reutenauer (1988) for a general reference on (rational) formal power series. Theorems 13.3 and 13.4 are classical results. What is 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);
- — our use of infinite weighted automata (as in the example of $\mathscr{T}(tan)$).

Part of all this can already be found in Rutten (2000a). New with respect to the latter is the application in Section 14 to coinductive counting. More about coinductive counting with weighted automata can be found in Rutten (2003), which also contains a coinductive treatment of continued fractions. Another approach to counting is the categorical theory of species (Bergeron *et al.* 1998). 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 (Gunawardena 1998)). 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 $\mathbb{R} \times (1 + (-))$.

Appendix.

In this appendix we give a more general theorem on the unique existence of the solution of systems of behavioural differential equations. Let $\Sigma = \{f, g, \ldots\}$ be a set of function symbols with arities $r_f, r_g, \ldots$, 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 $\sigma_1, \ldots, \sigma_r$, $\sigma_1', \ldots, \sigma_r'$, and $\sigma_1(0), \ldots, \sigma_r(0)$ (the latter real numbers considered as constant streams). And consider a real-valued function $h_f : \mathbb{R}^r \to \mathbb{R}$.

Theorem A.1. There is a unique solution to the following system of behavioural differential equations (one for each $f \in \Sigma$):

$$f(\sigma_1, \ldots, \sigma_r)' = t_f, \qquad f(\sigma_1, \ldots, \sigma_r)(0) = h_f(\sigma_1(0), \ldots, \sigma_r(0)),$$

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

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

$$(\sigma \times \tau)' = (\sigma' \times \tau) + (\sigma(0) \times \tau'), \qquad (\sigma \times \tau)(0) = \sigma(0) \times \tau(0)$$

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 of 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 $\mathbb{R}^\omega$ that assigns to each *syntactic term* $f(\sigma_1, \ldots, \sigma_r)$ 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 $(\sigma_1, \ldots, \sigma_r)$. For more details, refer to Rutten (2000a), where this construction is carried out fully for (a subset of) the operators of stream calculus. For a more general treatment of this type of coinductive definition, see also Lenisa (1998; 1999) and Bartels (2000). □

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 and Franck van Breugel for detailed comments on a draft of this paper. I am also grateful to the following 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

Aczel, P. (1988) *Non-well-founded sets*, CSLI Lecture Notes **14**, Center for the Study of Languages and Information, Stanford.

Aczel, P. and Mendler, N. (1989) A final coalgebra theorem. In: Pitt, D. H., Ryeheard, D. E., Dybjer, P., Pitts, A. M. and Poigne, A. (eds.) Proceedings category theory and computer science. *Springer-Verlag Lecture Notes in Computer Science* **389** 357–365.

Bartels, F. (2000) Generalised coinduction. Report SEN-R0043, CWI, 2000. Available at `www.cwi.nl`. (Extended abstract in Proceedings CMCS 2001. *Electronic Notes in Theoretical Computer Science* **44** (1), Elsevier.)

Bergeron, F., Labelle, G. and Leroux, P. (1998) Combinatorial species and tree-like structures. *Encyclopedia of Mathematics and its Applications* **67**, Cambridge University Press.

Birkhoff, G. and Rota, G.-C. (1978) *Ordinary differential equations* (third edition), John Wiley and Sons.

Berstel, J. and Reutenauer, C. (1988) Rational series and their languages. *EATCS Monographs on Theoretical Computer Science* **12**, Springer-Verlag.

Brzozowski, J. A. (1964) Derivatives of regular expressions. *Journal of the ACM* **11** (4) 481–494.

Comtet, L. (1974) *Advanced combinatorics*, D. Reidel Publishing Company.

Conway, J. H. (1971) *Regular algebra and finite machines*, Chapman and Hall.

Flajolet, P. and Sedgewick, R. (1993) The average case analysis of algorithms: Counting and generating functions. Research Report 1888, INRIA Rocquencourt. (116 pages.)

Flajolet, P. and Sedgewick, R. (2001) Analytical combinatorics: Functional equations, rational and algebraic functions. Research Report 4103, INRIA Rocquencourt. (98 pages.)

Graham, R. L., Knuth, D. E. and Patashnik, O. (1994) *Concrete mathematics* (second edition), Addison-Wesley.

Gunawardena, J. (1998) *Idempotency*, Publications of the Newton Institute, Cambridge University Press.

Jacobs, B. and Rutten, J. (1997) A tutorial on (co)algebras and (co)induction. *Bulletin of the EATCS* **62** 222–259. (Available at `www.cwi.nl/~janr.`)

Karczmarczuk, J. (1997) Generating power of lazy semantics. *Theoretical Computer Science* **187** 203–219.

Karczmarczuk, J. (2000) Lazy processing and optimization of discrete sequences. In: *Proceedings of the JFLA'2000*. (In French.)

Lenisa, M. (1998) *Themes in final semantics*, Ph.D. thesis, University of Udine, Udine, Italy.

Lenisa, M. (1999) From set-theoretic coinduction to coalgebraic coinduction. In: Jacobs, B. and Rutten, J. J. M. M. (eds.) Proceedings of CMCS'99. *Electronic Notes in Theoretical Computer Science* **19**, Elsevier.

McIlroy, M. D. (1999) Power series, power serious. *Journal of Functional Programming* **9** 323–335.

McIlroy, M. D. (2001) The music of streams. *Information Processing Letters* **77** 189–195.

Mikusinski, J. (1983) *Operational calculus*, Pergamon press.

Milner, R. (1980) A Calculus of Communicating Systems. *Springer-Verlag Lecture Notes in Computer Science* **92**.

Milner, R. (1989) *Communication and Concurrency*, Prentice Hall.

Park, D. M. R. (1981) Concurrency and automata on infinite sequences. In: Deussen, P. (ed.) Proceedings 5th GI conference. *Springer-Verlag Lecture Notes in Computer Science* **104** 167–183.

Pavlović, D. and Escardó, M. (1998) Calculus in coinductive form. In: *Proceedings of the 13th Annual IEEE Symposium on Logic in Computer Science*, IEEE Computer Society Press 408–417.

Rutten, J. J. M. M. (1998) Automata and coinduction (an exercise in coalgebra). Report SEN-R9803, CWI, 1998. (Available at `www.cwi.nl`. Also in: Sangiorgi, D. and de Simone, R. (eds.) The proceedings of CONCUR '98. *Springer-Verlag Lecture Notes in Computer Science* **1466** 194–218.)

Rutten, J. J. M. M. (1999) Automata, power series, and coinduction: taking input derivatives seriously (extended abstract). Report SEN-R9901, CWI, 1999. (Available at `www.cwi.nl`. Also in: Wiedermann, J., van Emde Boas, P. and Nielsen, M. (eds.) The proceedings of ICALP '99. *Springer-Verlag Lecture Notes in Computer Science* **1644** 645–654.)

Rutten, J. J. M. M. (2000a) Behavioural differential equations: a coinductive calculus of streams, automata, and power series. Report SEN-R0023, CWI, 2000. (Available at `www.cwi.nl.` To appear in *Theoretical Computer Science*.)

Rutten, J. J. M. M. (2000b) Universal coalgebra: a theory of systems. *Theoretical Computer Science* **249** (1) 3–80.

Rutten, J. J. M. M. (2001) Elements of stream calculus (an extensive exercise in coinduction). In: Brooks, S. and Mislove, M. (eds.) Proceedings of MFPS 2001: Seventeenth Conference on the Mathematical Foundations of Programming Semantics. *Electronic Notes in Theoretical Computer Science* **45**, Elsevier Science Publishers 1–66. (Also available as report SEN-R0120 at `www.cwi.nl`. Accepted for publication in *Mathematical Structures in Computer Science*.)

Rutten, J. J. M. M. (2003) Coinductive counting with weighted automata. *Journal of Automata, Languages and Combinatorics* **8** (2) 319–352.

Sangiorgi, D. (1998) On the bisimulation proof method. *Mathematical Structures in Computer Science* **8** (5) 447–479.

Scheid, F. (1968) *Theory and problems of numerical analysis*, Schaum's outline series, McGraw-Hill.

Sneddon, I. N. (1972) *The use of integral transforms*, McGraw-Hill.

Stanley, R. P. (1997) *Enumerative Combinatorics* I, Cambridge Studies in Advanced Mathematics **49**, Cambridge University Press.

Stanley, R. P. (1999) *Enumerative Combinatorics* II, Cambridge Studies in Advanced Mathematics **62**, Cambridge University Press.

Wilf, H. S. (1994) *Generatingfunctionology*, Academic Press.