Defining context-free power series coalgebraically

Marcello M. Bonsangue[2,1], Jan Rutten[1,3], and Joost Winter[1]⋆

[1] Centrum Wiskunde & Informatica (CWI)
[2] LIACS – Leiden University
[3] Radboud University Nijmegen

Abstract. In this paper we present a coinductive definition of context free power series in terms of behavioural differential equations. We show that our coalgebraic approach provides a unified view on many, at first sight different, existing notions of algebraicity, and we apply our behavioural differential equations to produce a new proof for a classical result by Chomsky and Schützenberger, and a simple proof that the zip-operator of two algebraic streams is algebraic.

1 Introduction

In [WBR11], we gave a coalgebraic presentation of context-free languages, characterizing them by means of so-called systems of *behavioural differential equations*. These equations define languages by specifying their initial (or output) value, which signals the presence of the empty word in the language, together with their (Brzozowski) input derivatives: one derivative for each letter in the alphabet of the languages under consideration. The main result of [WBR11] was a precise definition of a format for such behavioural differential equations, together with the observation that a language is context-free if and only if it can be specified by means of a system of equations in this format.

In the present paper, we will generalize the approach above from languages to *formal power series*, for which we define a general notion of context-freeness, again in terms of a well-defined class of behavioural differential equations. Formal power series include as examples languages but also other types of systems such as streams and infinite binary trees. Next we will connect our coalgebraic notion of context-free power series to various existing notions of *algebraic* power series, and establish, for a large number of cases, equivalence of these notions.

More precisely, we will provide a general, coalgebraic notion of context-freeness and we will show the following existing notions to be instances thereof:

- Automatic sequences over a finite field $\mathbb{F}_q$.
- Algebraic power series (as zeros of polynomial equations) in a single variable, over a perfect field F.

⋆ Supported by the Dutch NWO project CoRE: Coinductive Calculi for Regular Expressions.

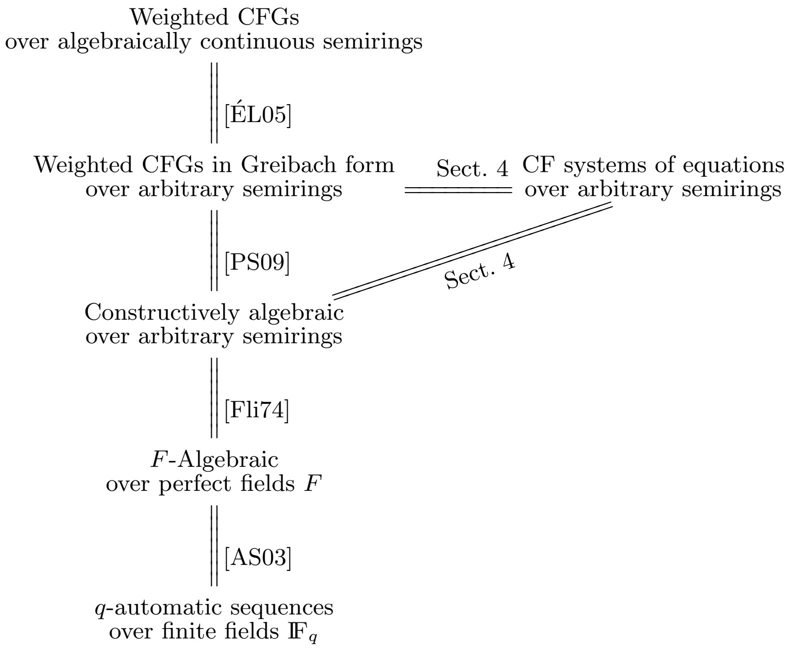


Fig. 1. An overview of the main equivalences

- 'Constructively' algebraic power series that are solutions to systems of equations in a specific format, over an arbitrary semiring S.
- Weighted context-free grammars in Chomsky and Greibach normal forms, over an arbitrary semiring S.
- Weighted context-free grammars in any form, over an algebraically complete semiring S.
- Weighted pushdown systems, over an arbitrary semiring S.

All these notions are equivalent to each other, and to the notion of 'context-free' power series defined in this article. We shall prove the latter through an easy correspondence between behavioural differential equations and algebraic systems in Greibach normal form. The other correspondences, some of which are not widely known, can all be found elsewhere in the literature (see Figure 1 for an overview of the various equivalences and for references to the literature).

The paper is organized as follows. In Section 2, we recall some of the basic definitions regarding algebraic structures; formal power series; and the coinductive calculus for streams and power series, fixing some notation as we go along. Section 3 will define our coalgebraic notion of a context-free power series, by means of behavioural differential equations. In Section 4, we will relate our coalgebraic notion of context-freeness to the various notions of algebraicity.

In addition to providing a unified view on many different but related notions of algebraicity, we shall illustrate, in Section 5, that the coalgebraic definition of context-freeness enables us to give new, transparent proofs of existing theorems. Two examples will be: a new proof for a result by Chomsky and Schützenberger (Section 5.2); and a new proof of various closure properties, such as the fact that the zip-operation applied to two algebraic streams again yields an algebraic stream (Section 5.1).

Acknowledgements: We would like to express our gratitude to Jeffrey Shallit, for pointing us towards the article by Michel Fliess [Fli74], in which the two notions of algebraicity are related, as well as to Jean Berstel, Christophe Reutenauer, Jean-Paul Allouche and Arto Salomaa. We thank the referees for providing constructive comments and help in improving this paper.

2 Preliminaries

2.1 Algebraic Structures

Recall that a *semiring* $(S, +, \cdot)$ consists of a set S, together with two binary operations $+$ and $\cdot$, and constants $0, 1 \in S$, satisfying the following conditions for all $s_1, s_2, s_3 \in S$:

$$\begin{array}{ll}
s_1 + (s_2 + s_3) = (s_1 + s_2) + s_3 & s_1 \cdot (s_2 \cdot s_3) = (s_1 \cdot s_2) \cdot s_3 \\
s_1 + 0 = s_1 = 0 + s_1 & s_1 \cdot 1 = s_1 = 1 \cdot s_1 \\
s_1 + s_2 = s_2 + s_1 & s_1 \cdot (s_2 + s_3) = s_1 \cdot s_2 + s_1 \cdot s_3 \\
s_1 \cdot 0 = 0 = 0 \cdot s_1 & (s_1 + s_2) \cdot s_3 = s_1 \cdot s_3 + s_2 \cdot s_3
\end{array}$$

A *ring* $(R, +, \cdot)$ is a semiring in which, for every element $r_1 \in R$, there is an additive inverse $r_2 \in R$, such that $r_1 + r_2 = 0 = r_2 + r_1$. A ring is called *commutative* if for all $r_1, r_2 \in R$, we have $r_1 \cdot r_2 = r_2 \cdot r_1$.

A *field* $(F, +, \cdot)$ is a commutative ring in which for every element $f_1 \in F$ with $f_1 \neq 0$, there is a multiplicative inverse f_2 such that $f_1 \cdot f_2 = 1 = f_2 \cdot f_1$.

For every field $(F, +, \cdot)$, there is a unique semiring homomorphism f from the semiring $(\mathbb{N}, +, \cdot)$ to F. If there are elements $n \in \mathbb{N}$ with $n \neq 0$ such that $f(n) = 0_F$, we say F has *characteristic* p if p is the smallest number with this condition; if there are no such elements, we say that F has characteristic 0. Whenever F has characteristic 0 then F is infinite, otherwise if $p \neq 0$ then p is a prime number and F is a finite field.

We say a field F is *perfect* when F has either characteristic 0, or when F has characteristic p and, for every $f \in F$, there is a $g \in F$ such that $f = g^p$ (here we inductively define $g^0 = 1$ and $g^{n+1} = g \cdot g^n$). Examples of perfect fields are the reals and of the rationals. The field of rational functions in one variable and with coefficients over a finite field is an example of a field that is not perfect.

2.2 Formal Power Series

In this subsection, we will introduce, using mostly classical notation, *formal power series* with coefficients in semirings S, and (noncommuting) variables in a finite alphabet A. Such series were originally introduced as a generalization of classical power series, abstracting away from the traditional interpretation as Taylor series of a function. Another way to look at formal power series is to regard them as generalizations of *formal languages*, or as a presentation of *weighted languages* over some semiring S.

For the remainder of this section, we fix a finite set A, called the *alphabet*, as well as a semiring $(S, +, \cdot)$ to denote the coefficients of the power series. Furthermore, recall that the set A^* of words over A can be regarded as a monoid, or more precisely, as the free monoid over A.

Given a semiring $(S, +, \cdot)$, and an alphabet A, a *formal power series with coefficients in S and noncommuting variables in A* is a function

$$\sigma : A^* \to S$$

assigning to each word in A^* an element of S. We let $S\langle\langle A\rangle\rangle$ denote the set of all power series with coefficients in S and variables in A. In the literature, the notation (σ, w) is widely used to denote the value of the series σ at the word w. In this paper, we will in addition often use the notation $\sigma(w)$ instead of (σ, w).

Given a formal power series σ, we say its *support* is the set of words $w \in A^*$ such that $\sigma(w) \neq 0$. We call a formal power series σ *polynomial* whenever its support is a finite set, or in other words, there are only finitely many words w such that $\sigma(w) \neq 0$. The set of all polynomial power series with coefficients in S and variables in A is denoted by $S\langle A\rangle$.

We will be using the fact that the set $S\langle\langle A\rangle\rangle$ of all power series is a *final coalgebra* of the functor functor $S \times (-)^A$ [Rut03]. The final coalgebra structure can be defined using a generalisation of the notion of *Brzozowski derivatives*, originally introduced for regular expressions and languages, as follows.

Given a coalgebra $(X, (o, \delta) : X \to S \times X^A)$, and an element $x \in X$, we write x_a for $\delta(x)(a)$, and call it the *a-derivative* of x; furthermore, we call $o(x)$ the *output value* of x. The set $S\langle\langle A\rangle\rangle$ can now be given a $S \times (-)^A$-coalgebra structure by defining, for $\sigma \in S\langle\langle A\rangle\rangle$, $a \in A$ and $w \in A^*$,

$$\sigma_a(w) := \sigma(a \cdot w)$$

and by setting, furthermore, $o(\sigma) := \sigma(\lambda)$, where λ is the empty word. One can easily prove that this turns $S\langle\langle A\rangle\rangle$ into a final coalgebra for the functor $S \times (-)^A$. Given any coalgebra for this functor, and an element t of its carrier set, we let $[\![t]\!]$ denote the value of t under the unique homomorphism of this coalgebra to $S\langle\langle A\rangle\rangle$.

Given a coalgebra $(X, (o, \delta))$ can furthermore generalize the notion of Brzozowski derivatives from derivatives to alphabet symbols to word derivatives, by setting x_λ for all $x \in X$ and inductively defining $x_{a \cdot w} = (x_a)_w$.

We also recall the following notion. For two $S \times (-)^A$-coalgebras $(X, (o_X, \delta_X))$ and $(Y, (o_Y, \delta_Y))$, a relation $\mathcal{R} \subseteq X \times Y$ is a *bisimulation* if $(x, y) \in \mathcal{R}$ implies $o_X(x) = o_Y(y)$ and, for all $a \in A$, $(x_a, y_a) \in \mathcal{R}$.

2.3 A coinductive calculus for streams and power series

We briefly recall a few basic facts from [Rut03], where a coinductive calculus of streams, languages and formal power series was introduced.

An algebraic structure can be conveniently given to the set $S\langle\langle A\rangle\rangle$ of formal power series by defining operators by means of so-called *behavioural differential equations.* For instance,the (element-wise) sum of two series σ and τ is defined as the unique series satisfying the following equations:

$$\begin{aligned} o(\sigma + \tau) &= o(\sigma) + o(\tau) \\ (\sigma + \tau)_a &= \sigma_a + \tau_a \end{aligned}$$

for all $a \in A$. Similarly, the following equations

$$\begin{aligned} o(\sigma \cdot \tau) &= o(\sigma) \cdot o(\tau) \\ (\sigma \cdot \tau)_a &= \sigma_a \cdot \tau \; + \; [o(\sigma)] \cdot \tau_a \end{aligned}$$

(with $[o(\sigma)] = (o(\sigma), 0, 0, 0, \ldots)$, i.e. the solution of the equation $o([o(\sigma)]) = o(\sigma)$ and $[o(\sigma)]_a = [0]$) uniquely determine the operation of *convolution product.*

If the alphabet A is a singleton set: $A = \{\mathcal{X}\}$, then $S\langle\langle A\rangle\rangle$ coincides with the set S^ω of streams over S. In that case, we write σ' for $\sigma_{\mathcal{X}}$ (and often also $\sigma(0)$ for $o(\sigma)$). If S is a field then for every series σ with $o(\sigma) \neq 0$ there exists an inverse, defined by the following equations:

$$\begin{aligned} o(\sigma^{-1}) &= o(\sigma)^{-1} \\ (\sigma^{-1})' &= -o(\sigma^{-1}) \cdot \sigma' \cdot \sigma^{-1} \end{aligned}$$

We will use the following shorthand notation:

$$\frac{\sigma}{\tau} := \sigma \cdot \tau^{-1}$$

For more details see [Rut03] and [Rut05].

3 Context-free power series

In this section, we will present a format for behavioural differential equations that generalizes the format used to characterize context-free languages, presented in [WBR11]. Here we will introduce a format for formal power series, which we will use to define the notion of *context-free* power series. As we will see later, the present notion coincides with an existing notion of context-free power series in the literature.

We start by presenting a syntax describing our format of behavioural differential equations. As before, let A be a finite set of *alphabet* symbols and let S be a semiring. Furthermore, let X be a (possibly infinite) set of *variables.* The variables $x \in X$ will represent power series in $S\langle\langle A\rangle\rangle$.

We call a behavioural differential equation (for formal power series) *well-formed* if it consists of (i) an equation $o(x) = s$, with $s \in S$, which specifies the output value of x; and (ii) an equation $x_a = t$, for each $a \in A$, with $t \in TX$, where TX is the following inductively defined set of terms:

$$TX \ni t ::= s \in S \,|\, x \in X \,|\, a \in A \,|\, t + t \,|\, t \cdot t$$

We say that a system of equations is *context-free* for X if it consists of one well-formed equation for each $x \in X$. Equivalently, a context-free system of equations over X can be viewed as a mapping $f : X \to S \times TX^A$, by defining $f = (o, \delta)$ with $\delta(x) = x_a$, as before.

We notice that for every set X, $S \times TX^A$ carries a suitable algebra structure so that we can interpret the above terms transforming a context-free system of equations $f : X \to S \times TX^A$ into a $S \times (-)^A$-coalgebra $\bar{f} : TX \to S \times TX^A$, where $\bar{f}$ is defined inductively as follows:

t	$o(t)$	t_a
x	$o(x)$	x_a (as specified by f)
s	s	0
b	0	if $b = a$ then 1 else 0
$u + v$	$o(u) + o(v)$	$u_a + v_a$
$u \cdot v$	$o(u) \cdot o(v)$	$u_a \cdot v + o(u) \cdot v_a$

Next we can combine a system of equations $f : X \to S \times TX^A$, its extension $\bar{f}$, and the final homomorphism into the final coalgebra $S\langle\langle A\rangle\rangle$, in the following diagram:

$$\begin{tikzcd}
X \arrow[r, hook, "i"'] \arrow[d, "f"] & TX \arrow[r, dashed, "{[\![\ \ ]\!]}"'] \arrow[dl, "\bar{f}"] & S\langle\langle A\rangle\rangle \arrow[d] \\
S \times TX^A \arrow[rr, dashed] & & S \times (S\langle\langle A\rangle\rangle)^A
\end{tikzcd}$$

A solution for our context-free system of equations is an assignment of variables $x \in X$ to power series s_x such that $s_x = [\![i(x)]\!] \in S\langle\langle A\rangle\rangle$ for all $x \in X$.

The above diagram shows that by every context-free system of equations has a unique solution, fixed by the definition extending f to $\bar{f}$, and the unique homomorphism from the coalgebra $(TX, \bar{f})$ into the final coalgebra.

The same solution can also be obtained by providing an appropriate semiring structure to $S \times TX^A$, which is equivalent to defining the extension from f to $\bar{f}$ in the sense that there is a unique semiring morphism from the free semiring (with variables in S, X, and A) TX to $S \times TX^A$ with the provided semiring structure, such that a) when applying the forgetful functor to this morphism, we obtain $\bar{f}$, and b) making the left part of the diagram commute. In this sense, the work here can be seen as an instance of the coalgebraic generalization of the powerset construction (see e.g. [SBBR10]).

By using the definition of $\bar{f}$, one can furthermore easily show that $[\![t_1 + t_2]\!]$ is the sum of $[\![t_1]\!]$ and $[\![t_2]\!]$, whereas $[\![t_1 \cdot t_2]\!]$ is the convolution product of $[\![t_1]\!]$ with $[\![t_2]\!]$.

Definition 1. *We call a formal power series over a semiring S and an alphabet A* context-free *if it is the solution of a context-free system of behavioural differential equations, for some finite set X of variables.*

As a first example of a context-free power series (over the semiring of the natural numbers and over a singleton alphabet), taken from [Rut02], consider the stream defined by the following equation:

$$o(x) = 1 \qquad x' = x \cdot x \tag{1}$$

Its solution is the stream of Catalan numbers

$$1, 1, 2, 5, 14, 42, 132, 429, 1430, \ldots$$

The nth element of this stream counts the number of well-bracketed words consisting of n pairs of opening and closing brackets. In Section 5.2, we will show how to derive the above equation from a context-free grammar representing pairs of brackets.

Another example of a context-free stream, using the finite field $\mathbb{F}_2$ (characterized by $1 + 1 = 0$) as underlying semiring, is defined by the following system of equations (where now $X = \{x, y, z, w\}$):

$$\begin{array}{ll} x(0) = 0 & x' = y \\ y(0) = 1 & y' = z \\ z(0) = 1 & z' = z + y + x \cdot x \cdot w \\ w(0) = 1 & w' = y \cdot w \end{array}$$

(here we write $x(0)$ etc. instead of $o(x)$). One can show that it generates the stream

$$0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, \ldots$$

which is the so-called Thue-Morse sequence. (This stream can also be characterized as the morphic sequence generated by the morphism with $p(0) = 01$ and $p(1) = 10$.) In Appendix A, a derivation of this system of equations will be given, and in Section 4.3, we will see yet another characterization of this sequence.

In some situations, it is convenient to deal with terms in a disjunctive normal form. We will later see that weighted grammars in weak Greibach normal form can be directly regarded as systems of equations in which every derivative is in disjunctive normal form.

Definition 2. *A system of equations $f : X \to S \times TX^A$ is in* disjunctive normal form *if for every $x \in X$ and $a \in A$, $x_a = t$ where t is a term of the following form:*

$$\begin{array}{l} t ::= c \,|\, c + t \\ c ::= a \cdot c \,|\, x \cdot c \,|\, s \quad (a \in A, x \in X, s \in S) \end{array}$$

We denote the subset of TX consisting of all terms that are in disjunctive normal form by DX. It is not very difficult to prove that every well-formed system of equations is equivalent to a system in disjunctive normal form.

4 Algebraicity and context-freeness

In this section, we will connect the notion of context-free power series defined above with several existing notions of *algebraic* power series, as well as with series definable by weighted grammars and q-automatic sequences.

We recall the following definition of *algebraicity*, given in [PS09], and originating from [Fli74].

Definition 3. *An S-*algebraic system *is a set of equations of the form*

$$x_i = p_i \qquad i \leq n$$

for some fixed $n \in \mathbb{N}$, where $p_i \in S\langle A \cup X \rangle$. Such a system is called proper *if, for all i, $(p_i, \lambda) = 0$, and for all $i, j \leq n$, $(p_i, x_j) = 0$. Furthermore, such a system is said to be in* Greibach normal *form if its support is contained in the set $A \cup AX \cup AXX$.*

A solution to an S-algebraic system consists of an assignment of power series in $S\langle A \rangle$ to the variables x_i's, such that when replacing the variables x_i's with the associated power series then all equations result valid [PS09]. In general a solution to an S-algebraic system neither needs to exists nor is unique. However, a system is proper, solutions exists and can be constructed as limit of an approximation sequence of power series starting from 0. Such solutions are called strong [PS09].

Definition 4. *A* strong solution *to an S-algebraic system is solution where for each power serie σ associated to a variable x of the system of equation it holds that $(\sigma, \lambda) = 0$. We call a formal power series σ* (constructively) S-algebraic *if it can be written as $s + \bar{\sigma}$, where $s \in S$ and $\bar{\sigma}$ is a strong solution to an S-algebraic system.*

In [PS09], the following results w.r.t. solutions to S-algebraic systems are presented:

- Every proper S-algebraic system has exactly one strong solution [PS09, Theorem 3.2].
- Every Boolean B-algebraic system (not necessarily proper) has a strong solution [PS09, Theorem 4].
- Any component of a strong solution to a proper S-algebraic system also occurs as a component of a strong solution to such a system in Greibach normal form [PS09, Theorem 3.2].

Given an S-algebraic system in Greibach normal form, it is easy to see that it is equivalent to a system of behavioral differential equations representing a context-free power series. Furthermore, if σ is context-free then also $\sigma + s$ is context-free, for any $s \in S$. It follows that any constructively S-algebraic power series also is context-free.

For the other direction, take any system of behavioural differential equations, and a variable y in this system representing a context-free power series. From this system, we construct a new system as follows: we replace each variable $x \in X$ by a variable $\bar{x}$ such that $(\bar{x}, \lambda) = 0$ and for all $w \in A^+$, $(\bar{x}, w) = (x, w)$. We do this, by setting $o(\bar{x}) = 0$ for each $\bar{x} \in \bar{X}$, and in each expression representing x_a, substituting $(\bar{y} + (y, \lambda))$ for each $y \in X$ in the corresponding expression for $\bar{x_a}$. It is easy to see that this new system of equations can be regarded as a proper S-algebraic system, so it follows that $\bar{y}$, and hence also y, is constructively S-algebraic.

All in all, we have proved the following.

Theorem 5. *A formal power series over a semiring S is S-context-free iff it is constructively S-algebraic.*

4.1 Weighted grammars and languages

Another way of viewing context-free power series is as the (weighted) languages generated by weighted context-free grammars [PS09]. A context-free grammar weighted over a semiring S consists of a set X of nonterminals, a set A of terminals, and an assignment

$$f : X \to \mathcal{P}_\omega(S \times (A \cup X)^*)$$

representing the traditional production rules of a context free grammar each associated with a weight in S. We write $x \xrightarrow{s} u$ whenever $(s, u) \in f(x)$, to say that x produces u with weight s.

We say that such a grammar is in weak Greibach normal form if, for every production $x \xrightarrow{s} u$, u is either the empty word λ, or of the form $a \cdot w$, with $a \in A$, and $w \in (A \cup X)^*$.

We can now extend weighted rules to weighted derivations as follows: when v is a sequence of terminals in A^*, x is a nonterminal in X, and w is a sequence of both terminals and nonterminals in $(A \cup X)^*$, we write

$$vxw \overset{s}{\Rightarrow} vuw$$

iff $x \xrightarrow{s} u$, and call this a *single leftmost derivation* (with weight s). A left-most derivation of word $w \in A^*$ from a nonterminal $x \in X$ is finite sequence of single leftmost derivation $x \overset{s_1}{\Rightarrow} \ldots \overset{s_n}{\Rightarrow} w$. There may be many different ways a word can be derived from a nonterminal x, therefore we can (try to) assign a weight s to every word w derived from x, denoted by $x \overset{s}{\Rightarrow}^* w$ as follows

$$s = \sum_{x \overset{s_1}{\Rightarrow} \ldots \overset{s_n}{\Rightarrow} w} \prod_{n} s_i \, .$$

The idea here is to take the sum over all derivations, and within each derivation, to take the product of the weight of each of the steps involved in the derivation. Note, however, that this sum may be an infinite sum which is not necessarily always defined. It is, however, always defined, whenever the grammar is in weak Greibach normal form. Grammars over the Boolean semiring or over the semiring of natural numbers extended with infitity always have an equivalent Greibach normal form [ÉL05].

Given a weighted grammar in Greibach normal form, we can construct a system of behavioural differential equations, with X, the set of nonterminals from the grammar, as set of variables for the system. We now can set $o(x) = s$ if there is a production rule $x \xrightarrow{s} \lambda$, and $o(x) = 0$ if no such rule exists; and furthermore, set

$$x_a = \sum_{x \xrightarrow{s} a \cdot w} s \cdot w$$

The equivalence between weighted grammars in weak Greibach normal form and well-formed systems of behavioural differential equations essentially follows from the following proposition, the easy proof of which is omitted.

Proposition 6. *For any word $w \in A^*$ have $o(x_w) = s$ iff $x \overset{s}{\Rightarrow}^* w$.*

4.2 *F*-algebraic streams

Next we consider yet another notion of algebraicity, which applies only to formal power series in one variable (here typically denoted by $\mathcal{X}$), that is, *streams*, with coefficients in a *field* F.

Definition 7. *An F-stream σ is called* F-algebraic *iff there exist polynomial F-streams $p_1, \ldots, p_n$, with at least one p_i not equal to zero, such that*

$$\sum_{i \leq n} p_i \sigma^i = 0.$$

The next proposition shows that under very mild conditions, we can constructively show that F-algebraic streams are context-free.

Proposition 8. *Given any field F, let σ be an F-algebraic stream that is a solution to the equation $\sum_{i,j \leq m,n} f_{ij} \mathcal{X}^j \sigma^i = 0$ (recall that a polynomial p is of the form $\sum_{i \leq n} f_i \mathcal{X}^i$). Whenever $\sum_{i \leq m} f_{i0} (\sum_{k<i} o(\sigma)^k \sigma^{i-k-1}) \neq 0$, then σ is context-free (and hence also constructively F-algebraic).*

Proof. Note that

$$\sum_{i,j \leq m,n} f_{ij} \sigma^i \mathcal{X}^j = \sum_{i \leq m} f_{i0} \sigma^i + \sum_{i \leq m, 1 \leq j \leq n} f_{ij} \sigma^i \mathcal{X}^j = 0$$

where all $f_{ij} \in F$. Taking derivatives, we now obtain

$$\sigma' \sum_{i \leq m} f_{i0} (\sum_{k<i} o(\sigma)^k \sigma^{i-k-1}) + \sum_{i \leq m, 1 \leq j \leq n} f_{ij} \sigma^i \mathcal{X}^{j-1} = 0$$

and hence

$$\sigma' \sum_{i \leq m} f_{i0} (\sum_{k<i} o(\sigma)^k \sigma^{i-k-1}) = - \sum_{i \leq m, 1 \leq j \leq n} f_{ij} \sigma^i \mathcal{X}^{j-1}$$

When $\sum_{i \leq m} f_{i0}(\sum_{k<i} o(\sigma)^k \sigma^{i-k-1}) \neq 0$, we can rewrite this as

$$\sigma' = -\frac{\sum_{i \leq m, 1 \leq j \leq n} f_{ij} \sigma^i \mathfrak{X}^{j-1}}{\sum_{i \leq m} f_{i0}(\sum_{k<i} o(\sigma)^k \sigma^{i-k-1})}$$

from which we can easily eliminate the fraction by introducing a new variable τ representing $(\sum_{i \leq m} f_{i0}(\sum_{k<i} o(\sigma)^k \sigma^{i-k-1}))^{-1}$.

Leaving out some further intermediate steps, we obtain the following system of behavioural differential equations in two variables, with $o(\sigma)$ given:

$$\begin{aligned}
\sigma' &= -(\sum_{i \leq m, 1 \leq j \leq n} f_{ij} \sigma^i \mathfrak{X}^{j-1}) \cdot \tau \\
o(\tau) &= (\sum_{i \leq m} f_{i0} \cdot (i-1) \cdot o(\sigma)^i)^{-1} \\
\tau' &= -o(\tau) \cdot \sigma' \cdot (\sum_{i \leq m} f_{i0}(\sum_{k \leq i-1} o(\sigma)^k (\sum_{j \leq i-k-2} o(\sigma)^j \sigma^{i-j-k-2}))) \cdot \tau
\end{aligned}$$

There is the following, more general result by Fliess [Fli74], stating that the notions of algebraic and constructively algebraic (and hence context-free) coincide for any *perfect* field.

Proposition 9. *Let F be a perfect field. Then a stream σ over F is F-algebraic iff it is constructively F-algebraic.*

This result is not very well-known. Its proof [Fli74, Proposition 7] is not very easy and too long to repeat here. We see it as one of the challenges for future research to better understand this proof, possibly by rephrasing it into the language of the present setting, along the lines of Proposition 8.

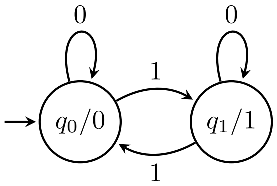
4.3 Automatic sequences

The theory of k-automatic sequences, developed in e.g. [AS03], is primarily concerned with streams over the set $\{0, \ldots, k-1\}$ that are characterized by so-called k-automata. Basically, a k-automaton is a deterministic Moore automaton with the set $\{0, \ldots, k-1\}$ as both input and output alphabets. However, differently from a Moore automaton, a k-automaton accepts streams. In fact, an infinite sequence σ is said to be k-automatic whenever a k-automaton exists, such that, for any $n \in \mathbb{N}$, taking the base k representation of n as input, the nth element $\sigma(n)$ is yielded as output. More formally:

Definition 10. *For any number $k \in \mathbb{N}$, a k-automaton is a deterministic automaton $A = (Q, T, q_0, o)$, with Q as set of states, $T : Q \times \{0, \ldots, k-1\} \to Q$ as transition function, an initial state q_0, and an output function $o : Q \to \{0, \ldots, k-1\}$.*

A sequence $\sigma(0)\sigma(1)\sigma(2)\ldots$ is called k-automatic whenever $\sigma(i)$ is the output obtained by taking the base k representation of the number i as input, for some k-automaton A.

As an example of an automatic sequence, consider the 2-automaton



which generates the Thue-Morse sequence. In [AS03], a large amount of information and results related to automatic sequences can be found.

It turns out that these k-automatic sequences, for prime numbers k, are exactly the $\mathbb{F}_k$-algebraic streams:

Theorem 11. *For any prime p, a sequence is p-automatic iff it is $\mathbb{F}_p$-algebraic.*

Proof. See [BR11, Theorem 4.1].

5 Applications of Context-Free Systems of Equations

In this section, we will see some examples of how the systems of context-free behavioural differential equations can be put to use, giving some clear and transparent proofs of existing theorems. The first example will provide a proof that the zip-operator, in all cases, preserves algebraicity, and the next example will consist of a new proof of a theorem by Chomsky and Schützenberger.

5.1 Zip preserves context-freeness

As an application of the above equivalences, we will now provide a newer and simpler version of [NR10, Proposition 18], stating that the zip operation preserves F-algebraicity over any field F. Recall that the zip operation can be defined coinductively as follows:

$$\begin{aligned} o(\mathrm{zip}(\sigma,\tau)) &= o(\sigma) \\ \mathrm{zip}(\sigma,\tau)' &= \mathrm{zip}(\tau,\sigma') \end{aligned}$$

We will now start by showing that, if σ is S-algebraic over a semiring S, then so are $\mathrm{zip}(\sigma,0)$ and $\mathrm{zip}(0,\sigma)$:

Proposition 12. *If σ is a S-algebraic stream over a semiring S, then* $\mathrm{zip}(\sigma,0)$ *and* $\mathrm{zip}(0,\sigma)$ *are also S-algebraic.*

Proof. Let σ be characterized by a system of equations in disjunctive normal form with X as set of variables. Now consider the following translation of terms in disjunctive normal form:

$$\begin{aligned} \mathcal{T}(s) &= s \quad (s \in S) \\ \mathcal{T}(\mathcal{X} \cdot c) &= \mathcal{X} \cdot (\mathcal{X} \cdot \mathcal{T}(c)) \\ \mathcal{T}(x \cdot c) &= \bar{x} \cdot \mathcal{T}(c) \quad (x \in X) \\ \mathcal{T}(c + t) &= \mathcal{T}(c) + \mathcal{T}(t) \end{aligned}$$

and consider a new system of equations, with $\bar{X} = \{\bar{x} \mid x \in X\}$ as new set of variables, and set, for $\bar{x} \in \bar{X}$: $o(\bar{x}) = o(x)$, and $\bar{x}' = \mathcal{X} \cdot \mathcal{T}(x')$.

Now consider the relation

$$\mathcal{R} = \{(\mathrm{zip}(u, 0), \mathcal{T}(u)) \mid u \in DX\} \cup \{(\mathrm{zip}(0, u), \mathcal{X} \cdot \mathcal{T}(u)) \mid u \in DX\}$$

which is easily seen to be a bisimulation (up-to bisimilarity and addition) between the old system augmented with the zip-operator, and the new system. Hence, it follows that $\mathrm{zip}(\sigma, 0)$ and $\mathrm{zip}(0, \sigma)$ are also S-algebraic.

We can now directly continue with the theorem:

Theorem 13. *Given a semiring S and streams σ and τ, $\mathrm{zip}(\sigma, \tau)$ is S-algebraic iff both σ and τ are S-algebraic.*

Proof. For the right to left direction, use

$$\mathrm{zip}(\sigma, \tau) = \mathrm{zip}(\sigma, 0) + \mathrm{zip}(0, \tau),$$

the previous result, plus the fact that addition preserves algebraicity. The left to right direction is also easy.

By using Fliess's Theorem 9, this result directly generalizes to F-algebraic streams over any perfect field F, and to automatic sequences.

A further easy generalization is using the $n-$zip of $n > 2$ different streams $\sigma_1, \ldots, \sigma_n$, defined as follows:

$$\begin{aligned} o(n\text{--}\mathrm{zip}(\sigma_1, \ldots, \sigma_n)) &= o(\sigma_1) \\ n\text{--}\mathrm{zip}(\sigma_1, \ldots, \sigma_n)' &= n\text{--}\mathrm{zip}(\sigma_2, \ldots, \sigma_n, \sigma_1) \end{aligned}$$

The following theorem can now be proved in a similar way as Theorem 13:

Theorem 14. *Given any semiring S and streams $\sigma_1, \ldots, \sigma_n$, $n-\mathrm{zip}(\sigma_1, \ldots, \sigma_n)$ is S-algebraic iff all σs are S-algebraic.*

5.2 Deriving systems of equations for streams

A theorem by Chomsky and Schützenberger states the following: if L is a context-free language admitting an unambiguous CFG, and a_k is the number of words of length k in L, then $\sum_{k \in \mathbb{N}} a_k x^k$ is $\mathbb{N}$-algebraic. We will provide a proof of this theorem using systems of behavioural differential equations, that also gives us a way to construct $\sum_{k \in \mathbb{N}} a_k x^k$ as a context-free power series.

The next theorem directly relates the degree of ambiguity of any CFG in weak Greibach normal form to a context-free formal power series using the semiring of natural numbers as underlying semiring. Recall that the *degree of ambiguity* of a word w w.r.t. a starting symbol x is equal to the number of distinct leftmost derivations $x \Rightarrow^* w$ [Nij80,PS09].

A lot of the argumentation and proofs in this section, even though not directly coinductive in the sense of using bisimulation techniques, still make use of the coalgebraic perspective. In fact the proofs of Propositions 15 and 16 are, at the heart, based on transformation of systems of behavioural differential equations to new equations.

Proposition 15. *Let (X, p) be a CFG in weak Greibach normal form over an alphabet A. We take the following system of equations, with $\mathbb{N}$ as underlying semiring, and X as set of variables. For each $x \in X$, $o(x) = 1$ if $x \to \lambda$ and 0 otherwise, and*

$$x_a = \sum_{u : x \to au} u$$

Now we claim the following: for each $t \in (A + X)^*$ and $w \in A^*$, the weight $o(t_w)$ of the w-derivative of t is equal to the number of distinct leftmost derivations $t \Rightarrow^* w$.

Proof. We proceed by induction on w.

Base step: if $w = \lambda$, then the number of distinct leftmost derivations $t \Rightarrow^* w$ is 1 iff every symbol in t is a variable x with a rule $t \to \lambda$, and 0 otherwise. But this corresponds precisely to $o(t_w) = o(t)$.

Induction step: say $w = a \cdot w'$, and the theorem holds for all v with $|v| < |w|$. We distinguish three cases now.

1. $t = a \cdot t'$. Then the number of distinct leftmost derivations $t \Rightarrow^* w$ is equal to the number of distinct leftmost derivations $t' \Rightarrow^* w'$. But also $o(t_w) = o((a \cdot t')_{a \cdot w'}) = o(t'_{w'})$. Now apply the inductive hypothesis.
2. $t = b \cdot t'$ with $b \neq a$. Then the number of distinct derivations $t \Rightarrow^* w$ is zero, and the same holds for $o(t_w)$.
3. $t = x \cdot t'$. Then the number of distinct leftmost derivations $x \cdot t' \Rightarrow^* a \cdot w'$ is equal to

$$\sum_{v : x \to av} \#(v \cdot t' \Rightarrow^* w') + (\textbf{If } x \to \lambda \textbf{ then } \#(t' \Rightarrow^* a \cdot w') \textbf{ else } 0)$$

where $\#(t \Rightarrow^* w)$ is the number of distinct leftmost derivations of w from t (if t is a variable x taken as a starting symbol, then this is the degree of ambiguity of w [Nij80,PS09]). Also,

$$\begin{aligned} o((x \cdot t')_{(a \cdot w')}) &= o(((x \cdot t')_a)_{w'} \\ &= o((x_a \cdot t' + o(x) \cdot t_a)_{w'}) \\ &= o((\sum_{v : x \to av} (v \cdot t') + o(x) \cdot t'_a)_{w'}) \\ &= \sum_{v : x \to av} o((v \cdot t')_{w'}) + o(x) \cdot o(t'_{a \cdot w'}) \end{aligned}$$

Now apply the inductive hypothesis.

Proposition 16. *Take a system of equations as constructed in Proposition 15. Now construct a new system of equations, representing a stream over $\mathbb{N}$, as follows: for each variable x in the old system, we let the new system have a corresponding variable $\bar{x}$.*

We now introduce a translation of terms with the following inductive definition: $\mathcal{T}(n) = n$ for $n \in \mathbb{N}$, $\mathcal{T}(x) = \bar{x}$ for $x \in X$, $\mathcal{T}(a) = \mathcal{X}$ for $a \in A$, $\mathcal{T}(t + u) = \mathcal{T}(t) + \mathcal{T}(u)$, and $\mathcal{T}(t \cdot u) = \mathcal{T}(t) \cdot \mathcal{T}(u)$.

Using this translation, we now set $o(\bar{x}) = o(x)$, and $\bar{x}' = \sum_{a \in A} \mathcal{T}(x_a)$, and make the following claim: for all terms t in the old system, and for all $n \in \mathbb{N}$

$$o(\mathcal{T}(t)^{(n)}) = \sum_{w : |w| = n} o(t_w)$$

Proof. By induction on n.

Base case: if $n = 0$, the only w s.t. $|w| = n$ is λ, so the claim now amounts to $o(\mathcal{T}(t)) = o(t)$, which can easily be seen by induction on t.

Inductive step: assume that the claim holds for all $m < n$. For n, use induction on terms. If $t = x$, then (as $\mathcal{T}(x) = \bar{x}$) the claim amounts to

$$o(\bar{x}^{(n)}) = \sum_{w : |w| = n} o(x_w)$$

But on one hand we now have

$$\begin{aligned} o(\bar{x}^{(n)}) &= o((\bar{x}')^{(n-1)}) \\ &= o((\sum_{a \in A} \mathcal{T}(x_a))^{(n-1)}) \\ &= \sum_{a \in A} o(\mathcal{T}(x_a)^{(n-1)}) \end{aligned}$$

and on the other hand we have

$$\sum_{w : |w| = n} o(x_w) = \sum_{a \in A} \sum_{w : |w| = n-1} o((x_a)_w)$$

and the desired equality now follows by applying the inductive hypothesis.

In the case where $t = u \cdot v$, we make use of the equalities

$$(u \cdot v)^{(n)} = \sum_{k = 0 \dots n} o(u^{(k)}) \cdot v^{(n-k)}$$

and

$$(u \cdot v)_w = \sum_{w = pq} o(u_p) \cdot v_q$$

Now observe

$$\begin{aligned} o(\mathcal{T}(u \cdot v)^{(n)}) &= o((\mathcal{T}(u) \cdot \mathcal{T}(v))^{(n)}) \\ &= o(\sum_{k = 0 \dots n} o(\mathcal{T}(u)^{(k)}) \cdot \mathcal{T}(v)^{(n-k)}) \\ &= \sum_{k = 0 \dots n} o(\mathcal{T}(u)^{(k)}) \cdot o(\mathcal{T}(v)^{(n-k)}) \end{aligned}$$

and

$$\begin{aligned} \sum_{w : |w| = n} o((u \cdot v)_w) &= \sum_{w : |w| = n} o(\sum_{w = pq} o(u_p) \cdot v_q) \\ &= \sum_{w : |w| = n} \sum_{w = pq} o(u_p) \cdot o(v_q) \\ &= \sum_{k = 0 \dots n} (\sum_{p : |p| = k} o(u_p))(\sum_{q : |q| = n-k} o(v_q)) \end{aligned}$$

Applying the (inner) inductive hypothesis now yields

$$o(\mathcal{T}(u \cdot v)^{(n)}) = \sum_{w:|w|=n} o((u \cdot v)_w),$$

as required.

The remaining base cases and inductive cases (w.r.t. the inner induction on terms) are easy.

Furthermore, it is known from e.g. [Nij80, Section 5.2.1] that, if a context-free language admits an unambiguous, non-left-recursive grammar (i.e. a grammar with no $a \in A$ and $w \in (A \cup X)^*$ such that $a \Rightarrow^+ aw$), then it also admits an unambiguous grammar in Greibach normal form.

Together (and sometimes combined with a bit of ingenuity), these theorems form the ingredients for deriving systems of equations for a specification of certain sequences, such as the Catalan numbers or the Schroeder numbers, as context-free streams.

As a first example, consider the Catalan numbers. It is well-known that the nth Catalan number corresponds to the number of ways to combine n pairs of matching brackets. An unambiguous CFG in (weak) Greibach normal form representing matching pairs of brackets is

$$x \to axbx \mid 1$$

which corresponds to the system of equations

$$o(x) = 1 \qquad x_a = xbx$$

We can transform this into a context-free specification of a stream, the even elements of which are the Catalan numbers, by replacing *every* alphabet symbol by the formal variable $\mathcal{X}$:

$$o(\sigma) = 1 \qquad \sigma' = \sigma \cdot \mathcal{X} \cdot \sigma$$

Now, because multiplication in streams is commutative, we can replace $\sigma \cdot \mathcal{X} \cdot \sigma$ by $\mathcal{X} \cdot \sigma \cdot \sigma$.

Now, if we remove the remaining $\mathcal{X}$ from the system of equations we obtain a new stream the elements of which are exactly the even elements of the previous stream, that is, exactly the Catalan numbers. That this is the case is easily seen by providing a bisimulation, relating the system

$$o(\sigma) = 1 \qquad \sigma' = \mathcal{X} \cdot \sigma \cdot \sigma$$

to $\text{zip}(\tau, \upsilon)$, where τ is the stream of the Catalan numbers as defined earlier in (1), and υ is a stream of 0's:

$$o(\tau) = 1 \qquad \tau' = \tau \cdot \tau \qquad o(\upsilon) = 0 \qquad \upsilon' = \upsilon$$

6 Conclusions and future work

We have presented a new, coalgebraic definition of *context-free* power series, by means of well-formed systems behavioural differential equations. In addition, we have shown this new definition has various existing notions as instances, including *context-free languages*, *algebraic power series*, *automatic sequences*, and *weighted context-free languages*.

Behavioural differential equations offer a very general and elegant mechanism for the definition of streams, languages and more generally formal power series. For one thing, they allow the use of coinduction when proving equalities and general laws. Further, by restricting the shape or format of these differential equations, one can characterize different classes of solutions. We see our definition of context-free power series as an interesting new instance of this more general phenomenon. (Other existing examples include *linear* behavioural differential equations, used for instance to define rational streams.)

It has also been an interesting exercise to establish and find in the literature the various results on how the different notions, such as context-free and algebraic, are related. One of the surprises was the theorem by Fliess on the equivalence of F-algebraic and constructively algebraic streams (cf. the acknowledgements).

It remains interesting to see how much further we can carry this process of finding coalgebraic proofs of language-theoretic theorems. Still on our list is a more transparent proof of Fliess' original theorem. A possible new direction is to see what can be done by varying the coinductively defined operators: for example, by adding the Hadamard product as a primitive, it is known that we will reach a class of power series beyond the algebraic/context-free class described in this paper.

References

[AS03] Jean-Paul Allouche and Jeffrey O. Shallit. *Automatic Sequences – Theory, Applications, Generalizations.* Cambridge University Press, 2003.

[BR11] Jean Berstel and Christophe Reutenauer. *Noncommutative Rational Series with Applications.* Cambridge University Press, 2011.

[ÉL05] Zoltán Ésik and Hans Leiß. Algebraically complete semirings and Greibach normal form. *Annals of Pure and Applied Logic*, 133(1-3):173–203, 2005.

[Fli74] Michel Fliess. Sur divers produits de sries formelles. *Bulletin de la S.M.F.*, 102:181–191, 1974.

[Nij80] Anton Nijholt. *Context-Free Grammars: Covers, Normal Forms, and Parsing*, volume 93 of *Lecture Notes in Computer Science.* Springer Verlag, Berlin, 1980.

[NR10] Milad Niqui and Jan J.M.M. Rutten. Sampling, splitting and merging in coinductive stream calculus. In Claude Bolduc, Jules Desharnais, and Béchir Ktari, editors, *Mathematics of Program Construction*, volume 6120 of *Lecture Notes in Computer Science*, pages 310–330. Springer, 2010.

[PS09] Ion Petre and Arto Salomaa. *Handbook of Weighted Automata*, chapter Algebraic systems and pushdown automata, pages 257–289. Springer, 1st edition, 2009.

[Rut02] Jan J.M.M. Rutten. Coinductive counting: bisimulation in enumerative combinatorics. *Electr. Notes Theor. Comput. Sci.*, 65(1):286–304, 2002.

[Rut03] Jan J.M.M. Rutten. Behavioural differential equations: a coinductive calculus of streams, automata, and power series. *Theoretical Computer Science*, 308(1-3):1–53, 2003.

[Rut05] Jan J.M.M. Rutten. A coinductive calculus of streams. *Mathematical Structures in Computer Science*, 15(1):93–147, 2005.

[WBR11] Joost Winter, Marcello M. Bonsangue, and Jan J.M.M. Rutten. Context-free languages, coalgebraically. In Andrea Corradini, Bartek Klin, and Corina Cîrstea, editors, *CALCO*, volume 6859 of *Lecture Notes in Computer Science*, pages 359–376. Springer, 2011.

[SBBR10] Alexandra Silva, Filippo Bonchi, Marcello Bonsangue, Jan J. M. M. Rutten. Generalizing the powerset construction, coalgebraically. In Kamal Lodaya, Meena Mahajan, editors, *FSTTCS 2010*, volume 8 of *LIPIcs*, pages 272–283.

A A concrete derivation of the context-freeness of Thue-Morse

In this appendix we find a specification of the Thue-Morse sequence as a context-free stream. We start from the following polynomial:

$$\mathcal{X} + (1 + \mathcal{X}^2)\sigma + (1 + \mathcal{X} + \mathcal{X}^2 + \mathcal{X}^3)\sigma^2$$

It is known that this polynomial has exactly two zeroes, namely the Thue-Morse sequence $(0, 1, 1, 0, 1, 0, \ldots)$ and its inverse $(1, 0, 0, 1, 0, 1, \ldots)$.[AS03, Example 12.1.2]

We will now derive a system of behavioral equations for the Thue-Morse sequence itself, i.e. the system with $\sigma(0) = 0$, $\sigma(1) = \sigma(2) = 1$. The main idea of this process is to eliminate the presence of the formal variable $\mathcal{X}$ by taking derivatives.

We start from the equation

$$\mathcal{X} + (1 + \mathcal{X}^2)\sigma + (1 + \mathcal{X} + \mathcal{X}^2 + \mathcal{X}^3)\sigma^2 = 0$$

which can be rewritten as

$$\sigma + \sigma^2 + \mathcal{X}(1 + \sigma^2) + \mathcal{X}^2(\sigma + \sigma^2) + \mathcal{X}^3\sigma^2 = 0$$

and, taking derivatives on both sides, we now obtain

$$\sigma' + (\sigma^2)' + 1 + \sigma^2 + \mathcal{X}(\sigma + \sigma^2) + \mathcal{X}^2\sigma^2 = 0$$

Because $(\sigma^2)' = \sigma' \cdot \sigma + \sigma(0) \cdot \sigma' = \sigma' \cdot \sigma + 0 \cdot \sigma' = \sigma' \cdot \sigma$, this equation can be rewritten as

$$\sigma' + \sigma' \cdot \sigma + 1 + \sigma^2 + \mathcal{X}(\sigma + \sigma^2) + \mathcal{X}^2\sigma^2 = 0 \tag{2}$$

Taking derivatives from (2) another time, we obtain

$$\sigma'' + (\sigma' \cdot \sigma)' + (\sigma^2)' + \sigma + \sigma^2 + \mathcal{X}\sigma^2 = 0$$

from which, using the earlier simplification of $(\sigma^2)'$, and the equality $(\sigma' \cdot \sigma)' = \sigma'' \cdot \sigma + \sigma'(0) \cdot \sigma' = \sigma'' \cdot \sigma + \sigma(1) \cdot \sigma' = \sigma'' \cdot \sigma + \sigma'$, we obtain

$$\sigma'' + (\sigma'' \cdot \sigma + \sigma') + \sigma' \cdot \sigma + \sigma + \sigma^2 + \mathcal{X}\sigma^2 = 0 \tag{3}$$

Taking derivatives from (3) yet another time, we obtain

$$\sigma''' + (\sigma'' \cdot \sigma)' + \sigma'' + (\sigma' \cdot \sigma)' + \sigma' + (\sigma^2)' + \sigma^2 = 0$$

which can be simplified, using the earlier equalities and $(\sigma'' \cdot \sigma)' = \sigma''' \cdot \sigma + \sigma(2) \cdot \sigma' = \sigma''' \cdot \sigma + \sigma'$, to

$$\sigma''' + (\sigma''' \cdot \sigma + \sigma') + \sigma'' + (\sigma'' \cdot \sigma + \sigma') + \sigma' + \sigma' \cdot \sigma + \sigma^2 = 0. \tag{4}$$

From (4), we can cancel out two of the three occurrences of the term σ', giving

$$\sigma''' + \sigma''' \cdot \sigma + \sigma'' + \sigma'' \cdot \sigma + \sigma' + \sigma' \cdot \sigma + \sigma^2 = 0$$

which, using the laws of distribution, can be simplified to

$$(1 + \sigma)(\sigma''') + (1 + \sigma)(\sigma'' + \sigma') + \sigma^2 = 0$$

which (because in the Boolean field $a + b = 0$ implies $a = b$) is equivalent to

$$(1 + \sigma)(\sigma''') = (1 + \sigma)(\sigma'' + \sigma') + \sigma^2. \tag{5}$$

Now, because $(1 + \sigma)(0) = 1(0) + \sigma(0) = 1 + 0 = 1$ (and, in the Boolean field, 1 has an inverse, namely 1), the inverse $(1 + \sigma)^{-1}$ is well-defined, and, multiplying both sides from (5) by this inverse we obtain

$$\begin{aligned} \sigma''' &= \frac{(1 + \sigma)(\sigma'' + \sigma') + \sigma^2}{1 + \sigma} \\ &= \frac{(1 + \sigma)(\sigma'' + \sigma')}{1 + \sigma} + \frac{\sigma^2}{1 + \sigma} \\ &= \sigma'' + \sigma' + \frac{\sigma^2}{1 + \sigma} \end{aligned}$$

This equation can be turned into a system of behavioural differential equations, by letting the variables x, y, z, w represent the streams σ, σ', σ'' and $(1 + \sigma)^{-1}$ respectively:

$$\begin{array}{rc} x(0) = 0 & x' = y \\ y(0) = 1 & y' = z \\ z(0) = 1 & z' = z + y + x^2 w \\ w(0) = 1 & w' = yw \end{array}$$

Of these equations, the equations for w need some clarification: the inverse τ^{-1} of a stream τ is defined as

$$(\tau^{-1})(0) = (\tau(0))^{-1} \; (\tau^{-1})' = -\tau(0)^{-1} \cdot (\tau' \cdot \tau^{-1})$$

We now have

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

and

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

confirming that $w(0) = 1$ and $w' = yw$ are indeed the appropriate equations for w.