An exercise in coinduction: Moessner's theorem

M. Niqui [*]
Centrum Wiskunde & Informatica (CWI)
m.niqui@cwi.nl

J.J.M.M. Rutten [†]
Centrum Wiskunde & Informatica (CWI)
and Radboud University Nijmegen
janr@cwi.nl

Abstract

We present a coinductive proof of Moessner's theorem. This theorem describes the construction of the stream $(1^n, 2^n, 3^n, \ldots)$ (for $n \geq 1$) out of the stream of natural numbers by repeatedly dropping and summing elements. Our formalisation consists of a direct translation of the operational description of Moessner's procedure into the equivalence of - in essence - two functional programs. Our proof fully exploits the circularity that is present in Moessner's procedure and is more elementary than existing proofs. As such, it serves as a non-trivial illustration of the relevance and power of coinduction.

1. Introduction

It is well-known that if one drops from the stream of natural numbers

$$(1, 2, 3, \ldots)$$

every second element:

$$(1, 3, 5, \ldots)$$

and one forms of the resulting stream (of the odd natural numbers) the stream of its partial sums:

$$(1, 4, 9, \ldots)$$

then one has obtained the stream of all the natural numbers squared:

$$(1^2, 2^2, 3^2, \ldots)$$

The cubes of the natural numbers can be obtained in a similar fashion, by dropping from the stream of natural numbers every *third* element, yielding

$$(1, 2, 4, 5, 7, 8, \ldots)$$

then taking partial sums:

$$(1, 3, 7, 12, 19, 27, \ldots)$$

then dropping every second element:

$$(1, 7, 19, \ldots)$$

and finally taking partial sums again:

$$(1, 8, 27, \ldots)$$

In [3], Moessner described how the above procedure of repeatedly alternating a drop and a partial sum operation can be generalised to obtain the stream

$$(1^n, 2^n, 3^n, \ldots)$$

for any $n \geq 1$. A proof of the correctness of this procedure, which is known as Moessner's theorem, was given by Perron in [6]. An alternative proof and further generalisations were later provided by Paasche [5] and Salié [9]. All these proofs are based on a detailed bookkeeping of the elements of all the intermediate streams, and use nested inductions, involving binomial coefficients and falling factorial numbers. More details about these classical proofs can be found in a recent paper [1, 2] by Hinze, in which he has given a new proof of

[*] The research of Milad Niqui was supported by the Netherlands Organisation for Scientific Research (NWO).

[†] Corresponding author.

Moessner's theorem (and its generalisations), in a calculation style.

Here, we present yet another proof of Moessner's theorem, using coinduction. This definition and proof principle, which is one of the cornerstones of the theory of coalgebra [7], is dual to the well-known principle of mathematical induction. Induction is well-suited for finite and well-founded structures, such as the natural numbers and finite lists. In contrast, coinduction can be used for reasoning about infinite structures such as the streams of natural numbers above.

Recently [4], we have studied various operations on streams from a coinductive perspective, including the drop operators that are used in Moessner's construction. This has made it possible to give an elementary proof of Moessner's theorem using coinduction.

Our proof has three main characteristics, in which it differs from the existing proofs mentioned above. First, our formalisation of Moessner's procedure consists of a direct translation of the operational description of Moessner's procedure with which we started this paper. Secondly, the construction of a suitable stream bisimulation, which as usual constitutes the heart of a proof by coinduction, fully exploits the circularity that is present in our definition of both the stream of natural numbers and the drop and sum operators. Thirdly, our proof is elementary to a degree that we expect that it can be easily automated, as is more often the case with coinductive proofs. This is certainly true for every concrete instance of the theorem, for a given fixed n. But also for the general case, an automated proof using an interactive proof assistant should be feasible.

None of these characteristics are shared by the aforementioned classic proofs by Perron, Paasche and Salié. And although Hinze's proof does exploit some of the circularity involved, using corecursive definitions of the operators it uses, his formalisation is at a considerable distance from the original operational description of Moessner's procedure. Moreover, the proof by Hinze is, to be sure, very interesting and clever but also somewhat ad hoc and relatively complex.

At the same time, the proofs by Paasche, Salié and Hinze deal with both Moessner's original theorem and with the generalisations mentioned above. In contrast, our present proof deals with Moessner's original procedure only. And although we conclude our paper with a formalisation of a representative example of these generalisations, finding a proof by coinduction is left as future work.

Summarizing, we believe that the present paper has a contribution to make to the foundations of functional programming. Our formalisation of Moessner's theorem can be easily presented as stating the equivalence of two functional (say Haskell) programs: the first computing the procedure of alternatingly dropping and summing elements of a stream, the second describing the computation the stream of n-powers of the stream of natural numbers. Furthermore, proving this equivalence constitutes a non-trivial exercise in coinduction, which is a definition and proof principle that is fundamental for functional programming. Finally, the present exercise is also interesting from a coalgebraic perspective, as it is – in our experience – one of the most interesting and advanced illustrations to date of the power of coinduction.

2. Preliminaries

We define the set of all *streams* of natural numbers by

$$\mathbb{N}^\omega = \{\sigma \mid \sigma : \mathbb{N} \to \mathbb{N}\}$$

We shall sometimes write such streams as

$$\sigma = (\sigma(0), \sigma(1), \sigma(2), \ldots)$$

We call $\sigma(0)$ the *initial value* of σ and we call the remainder of the stream the *stream derivative* of σ, denoted by

$$\sigma' = (\sigma(1), \sigma(2), \sigma(3), \ldots)$$

We can view streams as states of an abstract machine, for which initial value and derivative together determine the *behaviour*: one can think of the initial value $\sigma(0)$ as an (initial) observation on σ; and when we take one single transition step in state σ, we reach the new state σ'.

Next we define various streams and stream functions by so-called *stream differential equations* [8]. In analogy to differential equations in classical mathematics, stream differential equations define streams by specifying their stream derivative and their initial value.

- The stream $\overline{n} = (n, n, n, \ldots)$, for every $n \in \mathbb{N}$, is given by the following stream differential equation:

  $$\overline{n}' = \overline{n}$$

  with initial value $\overline{n}(0) = n$.

- The element-wise *sum*

  $$\sigma+\tau = (\sigma(0)+\tau(0),\, \sigma(1)+\tau(1),\, \sigma(2)+\tau(2),\, \ldots)$$

  of two streams $\sigma,\ \tau \in \mathbb{N}^\omega$ can be specified by the following stream differential equation:

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

  with initial value

  $$(\sigma+\tau)(0) = \sigma(0) + \tau(0)$$

  (We use overloading: the same symbol is used for the sum of natural numbers and the sum of streams.)
- Using the operation of sum, we can specify the stream of *natural numbers* $\mathsf{nat} = (1, 2, 3, \ldots)$ by

  $$\mathsf{nat}' = \mathsf{nat} + \overline{1}$$

  with initial value $\mathsf{nat}(0) = 1$.
- The (element-wise) *Hadamard product*

  $$\sigma \odot \tau = (\sigma(0)\cdot\tau(0),\, \sigma(1)\cdot\tau(1),\, \sigma(2)\cdot\tau(2),\, \ldots)$$

  of two streams $\sigma,\ \tau \in \mathbb{N}^\omega$ satisfies

  $$(\sigma\odot\tau)' = \sigma' \odot \tau'$$

  with initial value

  $$(\sigma\odot\tau)(0) = \sigma(0)\cdot\tau(0)$$

  Often we simply write $\sigma\tau$ for $\sigma\odot\tau$. Also we define nat^n by $\mathsf{nat}^0 = \overline{1}$ and $\mathsf{nat}^{n+1} = \mathsf{nat} \odot \mathsf{nat}^n$.
- Scalar multiplication

  $$k\sigma = (k\cdot\sigma(0),\, k\cdot\sigma(1),\, k\cdot\sigma(2),\, \ldots)$$

  of a stream $\sigma \in \mathbb{N}^\omega$ with a natural number $k \in \mathbb{N}$ satisfies:

  $$(k\sigma)' = k\sigma'$$

  with initial value

  $$(k\sigma)(0) = k\cdot\sigma(0)$$
- For every $\sigma \in \mathbb{N}^\omega$, the stream

  $$\Sigma\,\sigma = (\sigma(0),\, \sigma(0)+\sigma(1),\, \sigma(0)+\sigma(1)+\sigma(2),\, \ldots)$$

  of *partial sums* of σ is defined by the following stream differential equation:

  $$(\,\Sigma\,\sigma\,)' = (\,\Sigma\,\sigma'\,) + \overline{\sigma(0)}$$

  with initial value

  $$(\,\Sigma\,\sigma\,)(0) = \sigma(0)$$
- We define *drop operators* D_k^i, for all $k \geq 2$ and $0 \leq i < k$, and for all $\sigma \in \mathbb{N}^\omega$, by the following system of stream differential equations:

  $$(\,D_k^{i+1}\,\sigma\,)' = D_k^i\,\sigma'$$
  $$(\,D_k^0\,\sigma\,)' = D_k^{k-2}\,\sigma''$$

  with initial values

  $$(\,D_k^{i+1}\,\sigma\,)(0) = \sigma(0)$$
  $$(\,D_k^0\,\sigma\,)(0) = \sigma'\,(0)$$

  The operator D_k^i repeatedly drops the i-th element of every block of k elements of the incoming stream (please note that we start counting the elements of streams with 0). For instance,

  $$D_3^1(\sigma) = (\sigma(0), \sigma(2), \sigma(3), \sigma(5), \sigma(6), \sigma(8), \ldots)$$
- It will be convenient to have one function symbol for the composition of a drop operator with the operator for partial sums. Therefore we define

  $$\Sigma_k^i = \Sigma \circ D_k^i$$

  These operators satisfy the following differential equations:

  $$(\,\Sigma_k^{i+1}\,\sigma)' = \Sigma_k^i\,\sigma' + \overline{\sigma(0)}$$
  $$(\,\Sigma_k^0\,\sigma)' = \Sigma_k^{k-2}\,\sigma'' + \overline{\sigma'\,(0)}$$

  with initial values

  $$(\,\Sigma_k^{i+1}\,\sigma)\,(0) = \sigma(0)$$
  $$(\,\Sigma_k^0\,\sigma)\,(0) = \sigma'\,(0)$$

(It is straightforward to prove that all of the stream differential equations mentioned above are well-defined, that is, have a unique solution. See [8] for more details on stream differential equations.)

In our proof of Moessner's theorem, we will use a few basic properties of the operators above, all of which are easily verified.

Proposition 2.1 *For all* $n, m \in \mathbb{N}$,

$$\overline{n+m} = \overline{n} + \overline{m}$$

For all $\sigma, \tau, \rho \in \mathbb{N}^\omega$,

$$\sigma \odot \overline{1} = \sigma \qquad \sigma\odot\tau = \tau\odot\sigma$$

$$\sigma \odot (\tau + \rho) = (\sigma \odot \tau) + (\sigma \odot \rho)$$

$$D_k^i(\sigma + \tau) = D_k^i(\sigma) + D_k^i(\tau)$$

$$\Sigma_k^i(\sigma + \tau) = \Sigma_k^i(\sigma) + \Sigma_k^i(\tau)$$

We use stream differential equations not only because they offer a very succinct and convenient way of specifying streams. Equally importantly, they also allow us to build *stream bisimulation relations*, which are defined in terms of stream derivatives and initial values. Stream bisimulations are the key ingredient of proofs by coinduction, as we will see shortly.

Definition 2.2 (stream bisimulation)

A relation $R \subseteq \mathbb{N}^\omega \times \mathbb{N}^\omega$ is a *(stream) bisimulation* if, for all $\sigma,\, \tau \in \mathbb{N}^\omega$,

$$(\sigma, \tau) \in R \;\Rightarrow\; \begin{cases} (1) & \sigma(0) = \tau(0) \quad \text{and} \\ (2) & (\sigma',\, \tau') \in R \end{cases}$$

Theorem 2.3 (coinduction proof principle)

For a stream bisimulation relation $R \subseteq \mathbb{N}^\omega \times \mathbb{N}^\omega$ and for all $\sigma,\, \tau \in \mathbb{N}^\omega$,

$$(\sigma,\, \tau) \in R \;\Rightarrow\; \sigma = \tau$$

Proof:

If R is a bisimulation relation, then one proves $\sigma(n) = \tau(n)$, for all $\sigma,\, \tau \in \mathbb{N}^\omega$ with $(\sigma,\, \tau) \in R$, by induction on $n \in \mathbb{N}$. QED

Example 2.4 We illustrate the use of the coinduction proof principle with a simple example. The *shuffle product* of two streams σ and τ is classically defined by

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

Alternatively and equivalently, the shuffle product can be defined by the following stream differential equation:

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

with initial value

$$(\sigma \otimes \tau)(0) = \sigma(0) \cdot \tau(0)$$

An advantage of this definition is that it avoids the use of binomial coefficients. Now let us look at two basic properties of the shuffle product:

$$(\sigma + \tau) \otimes \rho = (\sigma \otimes \rho) + (\tau \otimes \rho)$$

$$(\sigma \otimes \tau) \otimes \rho = \sigma \otimes (\tau \otimes \rho)$$

The first property is straightforward to prove. If we base a proof of the second property, associativity, on the classical definition, then we shall encounter a double summation of terms with binomial coefficients. However, if we base a proof on the stream differential equation above, then our reasoning is pleasantly free of binomial coefficients. To this end, we define $R \subseteq \mathbb{N}^\omega \times \mathbb{N}^\omega$ to be the smallest set such that

(i) for all σ, τ and ρ in $\mathbb{N}^\omega$,

$$\langle (\sigma \otimes \tau) \otimes \rho,\, \sigma \otimes (\tau \otimes \rho) \rangle \in R$$

(ii) for all $\langle \sigma_1, \sigma_2 \rangle \in R$ and $\langle \tau_1, \tau_2 \rangle \in R$,

$$\langle \sigma_1 + \tau_1,\, \sigma_2 + \tau_2 \rangle \in R$$

It is easy to prove that R is a bisimulation relation. The associativity of the shuffle product now follows by the coinduction proof principle, Theorem 2.3. QED

3. Moessner's theorem

Using the definitions from Section 2, we shall now formalise Moessner's construction. In the formulation below, we start Moessner's construction not with the stream of natural numbers but with the constant stream $\overline{1} = (1, 1, 1, \ldots)$. This is equivalent to the description given in the introduction because, as we shall see, $\Sigma\,\overline{1} = \mathsf{nat}$ whence $\Sigma_{n+1}^n\,\overline{1} = \mathsf{nat}$, for all $n \geq 1$.

Theorem 3.1 (Moessner's theorem) *For all $n \geq 1$,*

$$\Sigma_2^1\, \Sigma_3^2\, \cdots\, \Sigma_{n+1}^n\, \overline{1} = \mathsf{nat}^n$$

We note that the above formula is a direct translation of the operational description of Moessner's procedure, given in the introduction.

4. The proof: warming up

We shall first prove Moessner's theorem for $n = 1$ and $n = 2$. The proofs will be by coinduction and consist of the construction of a stream bisimulation relation. After that, it will be easy to define one (big) bisimulation relation for Moessner's theorem in its full generality, for all $n \geq 1$ at the same time.

4.1 Moessner theorem for $n = 1$

In order to prove

$$\Sigma_2^1\,\overline{1} = \mathsf{nat}$$

by coinduction, a first naive attempt at the definition of a suitable stream bisimulation $R \subseteq \mathbb{N}^\omega \times \mathbb{N}^\omega$ is to put

$$R = \{(\Sigma_2^1\,\overline{1},\, \mathsf{nat})\}$$

In order to check whether R is a stream bisimulation relation, we compute initial values on the left and the right, which are equal to 1. Thus R satisfies stream bisimulation property (1), of Definition 2.2. Computing stream derivatives gives

$$(\Sigma_2^1\,\overline{1})' = \Sigma_2^0\,\overline{1} + \overline{1}$$

and

$$\mathsf{nat}' = \mathsf{nat} + \overline{1}$$

We see that R is not closed under stream derivatives, and so does not satisfy stream bisimulation property (2). In order to ensure that R will be closed under stream derivatives, our second attempt at defining R is now as follows: let $R \subseteq \mathbb{N}^\omega \times \mathbb{N}^\omega$ be the smallest set satisfying

(i) $(\Sigma_2^1\,\overline{1},\, \mathsf{nat}) \in R$

(ii) $(\Sigma_2^0\,\overline{1},\, \mathsf{nat}) \in R$

(iii) for all $\sigma \in \mathbb{N}^\omega$, $(\sigma, \sigma) \in R$

(iv) for all $(\sigma_1, \tau_1) \in R$ and $(\sigma_2, \tau_2) \in R$,

$$(\sigma_1 + \sigma_2,\, \tau_1 + \tau_2) \in R$$

We note that we have ensured that

$$(\,(\Sigma_2^1\,\overline{1})',\; \mathsf{nat}'\,) = (\,\Sigma_2^0\,\overline{1} + \overline{1},\; \mathsf{nat} + \overline{1}) \in R$$

by clauses (ii), (iii) and (iv). Similarly, also

$$(\,(\Sigma_2^0\,\overline{1})',\; \mathsf{nat}'\,) = (\,\Sigma_2^0\,\overline{1} + \overline{1},\; \mathsf{nat} + \overline{1}) \in R$$

It follows that our new R is indeed closed under derivatives. Also, one easily checks that initial values left and right are equal, for all pairs in R. This shows that R is a stream bisimulation. It follows, by coinduction Theorem 2.3, that $\Sigma_2^1\,\overline{1} = \mathsf{nat}$. QED

4.2 Moessner theorem for $n = 2$

For a proof by coinduction of

$$\Sigma_2^1\,\Sigma_3^2\,\overline{1} = \mathsf{nat}^2$$

we will define a relation $R \subseteq \mathbb{N}^\omega \times \mathbb{N}^\omega$ such that

$$(\,\Sigma_2^1\,\Sigma_3^2\,\overline{1},\;\; \mathsf{nat}^2\,) \in R$$

and such that R is a stream bisimulation. As before, we investigate stream derivatives left and right and compute

$$\begin{aligned}(\Sigma_2^1\,\Sigma_3^2\,\overline{1})' &= \Sigma_2^0(\Sigma_3^2\,\overline{1})' + \overline{(\Sigma_3^2\,\overline{1})(0)}\\ &= \Sigma_2^0(\Sigma_3^1\,\overline{1} + \overline{1}) + \overline{1}\\ &= \Sigma_2^0\Sigma_3^1\,\overline{1} + \Sigma_2^0\,\overline{1} + \overline{1}\end{aligned}$$

(using Proposition 2.1 for the last equality). Also,

$$\begin{aligned}(\mathsf{nat}^2)' &= (\,\mathsf{nat} \odot \mathsf{nat}\,)'\\ &= \mathsf{nat}' \odot \mathsf{nat}'\\ &= (\mathsf{nat} + \overline{1}) \odot (\mathsf{nat} + \overline{1})\\ &= \mathsf{nat}(\mathsf{nat} + \overline{1}) + \mathsf{nat} + \overline{1}\end{aligned}$$

We make a (mental) note to include the following three pairs in R:

$$(\,\Sigma_2^0\Sigma_3^1\,\overline{1},\, \mathsf{nat}(\mathsf{nat} + \overline{1})\,),\; (\,\Sigma_2^0\,\overline{1},\, \mathsf{nat}\,),\; (\,\overline{1},\, \overline{1}\,) \in R$$

We recognize the latter two pairs from the proof of Moessner's theorem for the case $n = 1$, and we continue with the computation of the stream derivatives of the streams in the first pair. Skipping a few intermediate steps, in which again some of the properties from Proposition 2.1 are used, we find:

$$\begin{aligned}&(\,\Sigma_2^0\Sigma_3^1\,\overline{1}\,)'\\ &\quad = \Sigma_2^0\Sigma_3^1\,\overline{1} + \Sigma_2^0\,\overline{1} + \Sigma_2^0\,\overline{1} + \overline{1} + \overline{1}\end{aligned}$$

and

$$\begin{aligned}&(\,\mathsf{nat}(\mathsf{nat} + \overline{1})\,)'\\ &\quad = \mathsf{nat}(\mathsf{nat} + \overline{1}) + \mathsf{nat} + \mathsf{nat} + \overline{1} + \overline{1}\end{aligned}$$

Based on the above analysis of (repeated) stream derivatives, we come to the following definition: let $R \subseteq \mathbb{N}^\omega \times \mathbb{N}^\omega$ be the smallest set satisfying

(i) $(\Sigma_2^1\,\overline{1},\, \mathsf{nat}) \in R$ and $(\,\Sigma_2^1\,\Sigma_3^2\,\overline{1},\;\; \mathsf{nat}^2\,) \in R$

(ii) $(\Sigma_2^0\,\overline{1},\, \mathsf{nat}) \in R$ and $(\,\Sigma_2^0\Sigma_3^1\,\overline{1},\, \mathsf{nat}(\mathsf{nat} + \overline{1})\,) \in R$

(iii) for all $\sigma \in \mathbb{N}^\omega$, $(\sigma, \sigma) \in R$

(iv) for all $(\sigma_1, \tau_1) \in R$ and $(\sigma_2, \tau_2) \in R$,

$$(\sigma_1 + \sigma_2,\ \tau_1 + \tau_2) \in R$$

One easily verifies that R is a stream bisimulation relation. It follows, by coinduction Theorem 2.3, that $\Sigma_2^1 \, \overline{1} = \mathsf{nat}$ and that $\Sigma_2^1 \, \Sigma_3^2 \, \overline{1} = \mathsf{nat}^2$. In other words, the above relation R proves Moessner's theorem for $n = 1$ and $n = 2$ at the same time. QED

5. The proof: general case

We shall now prove Moessner's Theorem 3.1, for all $n \geq 1$. For the proof, we define again a stream bisimulation relation, generalising the relations used previously, as follows: Let $R \subseteq \mathbb{N}^\omega \times \mathbb{N}^\omega$ be the smallest set satisfying

(i) for all $n \geq 1$,

$$(\,\Sigma_2^1 \, \Sigma_3^2 \, \cdots \, \Sigma_{n+1}^n \, \overline{1},\ \mathsf{nat}^n\,) \in R$$

(ii) for all $n \geq 1$,

$$(\,\Sigma_2^0 \, \Sigma_3^1 \, \cdots \, \Sigma_{n+1}^{n-1} \, \overline{1},\ \mathsf{nat}(\mathsf{nat} + \overline{1})^{n-1}\,) \in R$$

(iii) for all $\sigma \in \mathbb{N}^\omega$,

$$(\sigma, \sigma) \in R$$

(iv) for all $(\sigma_1, \tau_1) \in R$ and $(\sigma_2, \tau_2) \in R$,

$$(\sigma_1 + \sigma_2,\ \tau_1 + \tau_2) \in R$$

Next we will prove that R is a stream bisimulation. Moessner's theorem then follows by coinduction, Theorem 2.3.

In order to prove that the relation R is a bisimulation, we shall use the following facts.

Proposition 5.1 *For all* $n \geq 1$,

$$\mathsf{nat}^n(0) = 1$$

and

$$\begin{aligned} &(\mathsf{nat}^n)' \\ &\quad = \ \mathsf{nat}(\mathsf{nat} + \overline{1})^{n-1} \\ &\quad + \ \mathsf{nat}(\mathsf{nat} + \overline{1})^{n-2} \\ &\quad + \ \cdots \\ &\quad + \ \mathsf{nat}(\mathsf{nat} + \overline{1}) \\ &\quad + \ \mathsf{nat} + \overline{1} \end{aligned}$$

Proposition 5.2 *For all* $k, n \geq 1$,

$$(\,\Sigma_{k+1}^1 \, \Sigma_{k+2}^2 \, \cdots \, \Sigma_{k+n}^n \, \overline{1}\,)(0) = 1$$

and

$$\begin{aligned} &(\,\Sigma_{k+1}^1 \, \Sigma_{k+2}^2 \, \cdots \, \Sigma_{k+n}^n \, \overline{1}\,)' \\ &\quad = \ \Sigma_{k+1}^0 \, \Sigma_{k+2}^1 \, \cdots \, \Sigma_{k+n}^{n-1} \, \overline{1} \\ &\quad + \ \Sigma_{k+1}^0 \, \Sigma_{k+2}^1 \, \cdots \, \Sigma_{k+n-1}^{n-2} \, \overline{1} \\ &\quad + \ \cdots \\ &\quad + \ \Sigma_{k+1}^0 \, \Sigma_{k+2}^1 \, \overline{1} \\ &\quad + \ \Sigma_{k+1}^0 \, \overline{1} \ + \ \overline{1} \end{aligned}$$

Proposition 5.3 *For all* $n \geq 1$,

$$(\,\mathsf{nat}(\mathsf{nat} + \overline{1})^{n-1}\,)(0) = 2^{n-1}$$

and

$$\begin{aligned} &(\mathsf{nat}(\mathsf{nat} + \overline{1})^{n-1})' \\ &\quad = \ a_0^{n-1}\mathsf{nat}(\mathsf{nat} + \overline{1})^{n-1} \\ &\quad + \ a_1^{n-1}\mathsf{nat}(\mathsf{nat} + \overline{1})^{n-2} \\ &\quad + \ \cdots \\ &\quad + \ a_{n-2}^{n-1}\mathsf{nat}(\mathsf{nat} + \overline{1}) \\ &\quad + \ a_{n-1}^{n-1}(\mathsf{nat} + \overline{1}) \end{aligned}$$

where, for $0 \leq i \leq n-1$,

$$\begin{aligned} a_i^{n-1} \ = \ &\binom{n-1}{i} \\ &+ \cdots \\ &+ \binom{n-1}{1} \\ &+ \binom{n-1}{0} \end{aligned}$$

Proposition 5.4 *For all* $k, n \geq 1$,

$$(\,\Sigma_{k+1}^0 \, \Sigma_{k+2}^1 \, \cdots \, \Sigma_{k+n}^{n-1} \, \overline{1}\,)(0) = 2^{n-1}$$

and

$$\begin{aligned} &(\,\Sigma_{k+1}^0 \, \Sigma_{k+2}^1 \, \cdots \, \Sigma_{k+n}^{n-1} \, \overline{1}\,)' \\ &\quad = \ a_0^{n-1}\Sigma_{k+1}^{k-1} \, \Sigma_{k+2}^k \, \cdots \, \Sigma_{k+n}^{k+n-2} \, \overline{1} \\ &\quad + \ a_1^{n-1}\Sigma_{k+1}^{k-1} \, \Sigma_{k+2}^k \, \cdots \, \Sigma_{k+n-1}^{k+n-3} \, \overline{1} \\ &\quad + \ \cdots \\ &\quad + \ a_{n-2}^{n-1}\Sigma_{k+1}^{k-1} \, \Sigma_{k+2}^k \, \overline{1} \\ &\quad + \ a_{n-1}^{n-1}\,(\,\Sigma_{k+1}^{k-1} \, \overline{1} \ + \ \overline{1}\,) \end{aligned}$$

with the a_i's as above, in Proposition 5.3.

The proofs of Propositions 5.1, 5.2 and 5.3 are straightforward. The proof of Proposition 5.4 is by induction on n. In the induction step, one uses the following property of binomial coefficients: for all $n \geq 1$ and $0 \leq i \leq n-1$,

$$a_i^{n-1} = a_i^{n-2} + a_{i-1}^{n-3} + \cdots + a_0^{n-1-i}$$

Using these four propositions, one can easily show that the relation R, defined at the beginning of this section, is a stream bisimulation relation. Moessner's theorem, now for all $n \geq 1$, follows as before by coinduction. QED

6. Stream calculus

The coinductive proof of the previous section is what we see as the main contribution of this paper. In the present section, we want to give yet another proof of Moessner's theorem, which can be viewed as an equational version of the proof by coinduction.

We shall be using a bit of elementary stream calculus [8], of which the basics are briefly recalled first.

We define the set of all streams of real numbers by

$$\mathbb{R}^\omega = \{\sigma \mid \sigma : \mathbb{N} \to \mathbb{R}\}$$

For $r \in \mathbb{R}$, we define the constant stream

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

which we often denote again by r. Another constant stream is

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

The operation of *sum* is given, as before, by

$$(\sigma + \tau)(n) = \sigma(n) + \tau(n)$$

for $\sigma, \tau \in \mathbb{R}^\omega$ and $n \geq 0$. We shall also need yet another type of product, called the *(convolution) product*:

$$(\sigma \times \tau)(n) = \sum_{i=0}^{n} \sigma(i) \cdot \tau(n-i)$$

The convolution product is different from both the Hadamard product and the shuffle product that we defined previously (on the natural numbers).

Important – a new notational convention: Previously we wrote $\sigma\tau$ for the Hadamard product $\sigma \odot \tau$. In what follows, we shall reserve this notation for the convolution product:

$$\sigma\tau = \sigma \times \tau$$

Also we now define $\sigma^0 = [1]$ and $\sigma^{n+1} = \sigma \times \sigma^n$. We shall still want to use Hadamard powers of streams, for which we introduce a new notation. We define $\sigma^{h(0)} = \overline{1}$ and $\sigma^{h(n+1)} = \sigma \odot \sigma^{h(n)}$.

We continue with our summary of stream calculus. If $\sigma(0) \neq 0$ then the stream σ has a (unique) multiplicative inverse σ^{-1} in $\mathbb{R}^\omega$, satisfying $\sigma^{-1} \times \sigma = [1]$. As usual, we shall often write $1/\sigma$ for σ^{-1} and σ/τ for $\sigma \times \tau^{-1}$.

We call a stream $\pi \in \mathbb{R}^\omega$ *polynomial* if there are $k \geq 0$ and $a_i \in \mathbb{R}$ such that

$$\begin{aligned} \pi &= a_0 + a_1 X + a_2 X^2 + \cdots + a_k X^k \\ &= (a_0,\, a_1,\, a_2,\, \ldots,\, a_k,\, 0,\, 0,\, 0,\, \ldots) \end{aligned}$$

where we write $a_i X^i$ for $[a_i] \times X^i$ with X^i the i-fold convolution product of X with itself. A stream $\rho \in \mathbb{R}^\omega$ is *rational* if it is the quotient $\rho = \sigma/\tau$ of two polynomial streams σ and τ with $\tau(0) \neq 0$. For instance, the following steams are rational:

$$\frac{1}{1-2X} = (1, 2, 2^2, 2^3, \ldots)$$

$$\frac{X}{(1-X)^2} = (0, 1, 2, 3, \ldots)$$

One can compute a stream from its initial value and derivative by the so-called *fundamental theorem* of stream calculus [8]: for all $\sigma \in \mathbb{R}^\omega$,

$$\sigma = \sigma(0) + (X \times \sigma')$$

(writing $\sigma(0)$ for $[\sigma(0)]$). The fundamental theorem of stream calculus allows us to solve stream differential equations. For a trivial example, take

$$\sigma(0) = 1 \qquad \sigma' = \sigma$$

By the fundamental theorem, we have $\sigma = \sigma(0) + (X \times \sigma') = 1 + (X \times \sigma)$, which leads to the solution $\sigma = 1/1 - X$ (which happens to be equal to the stream $\overline{1}$).

In the remainder of this section, we shall apply the stream calculus above for yet another proof of Moessner's theorem. More specifically, we shall prove that both streams on the left-hand side and the right-hand side of Moessner's identity are rational, using the fundamental theorem together with the propositions from Section 5. Then Moessner's follows simply from the observation that these rational streams are equal.

6.1 A rational expression for $\Sigma_2^1\,\Sigma_3^2\,\cdots\,\Sigma_{n+1}^n\,\overline{1}$

For notational convenience, we introduce the following constants:

$$\begin{aligned} P_n &= \Sigma_2^1\,\Sigma_3^2\,\cdots\,\Sigma_{n+1}^n\,\overline{1} \\ Q_n &= \Sigma_2^0\,\Sigma_3^1\,\cdots\,\Sigma_{n+1}^{n-1}\,\overline{1} \end{aligned}$$

for all $n \geq 1$. For the numbers P_n, we have

$$\begin{aligned} & P_n \\ &= P_n(0) + (X \times P_n') \\ &= 1 + X \times (Q_n + Q_{n-1} + \\ &\qquad \cdots + Q_2 + Q_1 + \overline{1}) \end{aligned}$$

And for the numbers Q_n, we have

$$\begin{aligned} & Q_n \\ &= Q_n(0) + (X \times Q_n') \\ &= 2^{n-1} + X \times (a_0^{n-1} Q_n + a_1^{n-1} Q_{n-1} + \\ &\qquad \cdots + a_{n-2}^{n-1} Q_2 + a_{n-1}^{n-1}(Q_1 + \overline{1})) \end{aligned}$$

where the coefficients a_i^j are defined as in Proposition 5.3. As a consequence, we obtain the following recurrence relation for Q_n:

$$\begin{aligned} & Q_n \\ &= \frac{2^{n-1}}{1-X} + \frac{X}{1-X}\,(a_1^{n-1} Q_{n-1} + \\ &\qquad \cdots + a_{n-2}^{n-1} Q_2 + a_{n-1}^{n-1}(Q_1 + \overline{1})) \end{aligned}$$

This recurrence together with the above formula for P_n allows us to compute a closed rational expression for (both Q_n and) P_n, yielding the following formulae, for the first few values of n:

$$\begin{aligned} P_1 &= \frac{1}{(1-X)^2} \\ P_2 &= \frac{1+X}{(1-X)^3} \\ P_3 &= \frac{1+4X+X^2}{(1-X)^4} \\ P_4 &= \frac{1+11X+11X^2+X^3}{(1-X)^5} \\ P_5 &= \frac{1+26X+66X^2+26X^3+X^4}{(1-X)^6} \end{aligned}$$

(A general formula for P_n, for arbitrary $n \geq 1$, will be discussed below.)

6.2 A rational expression for $\mathsf{nat}^{h(n)}$

In a similar fashion, we are able to compute rational expressions for all the (Hadamard) powers of nat. We compute as follows:

$$\begin{aligned} & \mathsf{nat}^{h(n)} \\ &= (\mathsf{nat}^{h(n)})(0) + (X \times (\mathsf{nat}^{h(n)})') \\ &= 1 + X \times (\overline{1} + \mathsf{nat})^{h(n)} \\ &= 1 + X \times (\overline{1} + \binom{n}{1}\mathsf{nat}^{h(1)} + \\ &\qquad \cdots + \binom{n}{n-1}\mathsf{nat}^{h(n-1)} + \mathsf{nat}^{h(n)}) \end{aligned}$$

This implies the following recurrence relation:

$$\begin{aligned} & \mathsf{nat}^{h(n)} \\ &= \frac{1}{1-X} + \frac{X}{1-X}(\mathsf{nat}^{h(0)} + \binom{n}{1}\mathsf{nat}^{h(1)} + \\ &\qquad \cdots + \binom{n}{n-1}\mathsf{nat}^{h(n-1)}) \end{aligned}$$

where we have replaced $\overline{1}$ by $\mathsf{nat}^{h(0)}$. It leads to the following rational expressions, again for the first few values of n:

$$\begin{aligned} \mathsf{nat}^{h(1)} &= \frac{1}{(1-X)^2} \\ \mathsf{nat}^{h(2)} &= \frac{1+X}{(1-X)^3} \\ \mathsf{nat}^{h(3)} &= \frac{1+4X+X^2}{(1-X)^4} \\ \mathsf{nat}^{h(4)} &= \frac{1+11X+11X^2+X^3}{(1-X)^5} \\ \mathsf{nat}^{h(5)} &= \frac{1+26X+66X^2+26X^3+X^4}{(1-X)^6} \end{aligned}$$

6.3 Yet another proof of Moessner's theorem

Because the rational expressions for P_1, P_2, etc. are equal to those for $\mathsf{nat}^{h(1)}$, $\mathsf{nat}^{h(2)}$, etc., we have proved Moessner's theorem again, for each of these cases. For a general proof, for all $n \geq 1$ at the same time, we have to determine a general expression for arbitrary n, for both P_n and $\mathsf{nat}^{h(n)}$. To this end, we use the fact that there exists in the literature a *generating function* for the n-powers of the natural numbers. Such generating functions can be (almost literally) translated into an expression in stream calculus. Based on this, one can prove the following general identity, for all $n \geq 1$:

$$P_n \;=\; \sum_{m=0}^{n-1} A(n,m)\,\frac{X^m}{(1-X)^{n+1}} \;=\; \mathsf{nat}^{h(n)}$$

where $A(n,m)$ are the so-called *Eulerian numbers*, which are defined, for every $n \geq 1$ and $0 \leq m \leq n-1$, by the following recurrence relation:

$$\begin{array}{rcl} A(n,m) & = & (n-m)A(n-1.m-1) \\ & + & (m+1)A(n-1,m) \end{array}$$

All details are omitted here. The above identity constitutes yet another proof of Moessner's theorem.

7. Discussion

Here are what we see as the main constituents of our coinductive proof of Moessner's theorem.

- Streams are viewed as single entities.
- The coinduction proof principle, Theorem 2.3, says that in order to prove that two streams *are* the same, it suffices to show that they *behave* the same (since two streams have the same behaviour if they are related by a bisimulation). For streams, in other words,

  being is doing
- Showing that two streams behave the same (and hence are equal) is particularly easy when their behaviour is *circular*. This makes it possible to construct finite or (using induction) finitary bisimulation relations.
- For the proof of Moessner's theorem, the circularity involved is expressed by the stream differential equations for the operations of partial summation and dropping, on the one hand, and the stream of natural numbers, on the other. To illustrate this for the natural numbers, we recall that

  $$\mathsf{nat}' = \mathsf{nat} + \overline{1}$$

  Here we have circular behaviour in that after one transition step of nat, we obtain a new state nat' that contains nat again as a summand.
- We arrived at the definition of the bisimulation relation R used in the proof of Moessner's theorem in a fairly standard way. First we constructed R for the cases of n equal to 1 and 2. For each of these cases, we included first the pair of streams that we wanted to prove equal (clause (i) of the definition of R). Then we computed their derivatives and observed that they consist of sums of the streams we started out with together with some new streams. The latter were added as new pairs, in clause (ii). Closing R under sums, in clause (iii) and including the identity relation, in clause (iv), then was sufficient to prove that R is a bisimulation. Having gone through this process for the first few values of n, the general definition of R emerged.

Our coinductive proof of Moessner's theorem, based on the construction of a bisimulation relation, is closely related to our second proof, based on rational expressions. For the definition of the bisimulation relation, we had to analyse the initial values and derivatives of P_n and $\mathsf{nat}^{h(n)}$, which resulted in the propositions of Section 5. Similarly, the recurrence relations that led to the rational expressions for P_n and $\mathsf{nat}^{h(n)}$ are based on these same propositions.

We see two main subjects for future research. First, we want to analyse the precise relationships between our present two proofs, on the one hand, and the proofs by Perron [6], Paasche [5] and Salié [9], and by Hinze [1], on the other.

Secondly, we want to try and apply our coinductive proof method to the following generalisation of Moessner's theorem, which is proved in [5, 9] and also analysed in [1]. A first contribution to a coinductive proof of this generalisation is already contained in our formulation of it here: we present its ingredients in terms of again a stream diferential equation, using the following slightly adapted version of the drop operator: for all $\sigma \in \mathbb{N}^\omega$ and all $k \geq 2$ and $0 \leq i < k$, we define by

$$\begin{array}{ll} (\,D_k^{i+1}\,\sigma)\,(0) \;=\; \sigma(0) & \quad (\,D_k^{i+1}\,\sigma\,)' \;=\; D_k^i\,\sigma' \\ (\,D_k^0\,\sigma)\,)(0) \;=\; \sigma'\,(0) & \quad (\,D_k^0\,\sigma\,)' \;=\; D_{k+1}^{k-1}\,\sigma'' \end{array}$$

The difference between this definition and the one in Section 2 lies in the value of $(\,D_k^0\,\sigma\,)'$, which changes the cycle of the drop operator from k to $k{+}1$. Using this new definition, we define an operation M on streams $\sigma \in \mathbb{N}^\omega$ of natural numbers by the following stream differential equation:

$$M(\sigma)(0) \;=\; \sigma(0) \qquad (M(\sigma))' \;=\; M(\,\Sigma \circ D_2^1(\sigma')\,)$$

The question with which we want to conclude the present paper is: to give a coinductive proof of the fact that

$$M(\bar{1}) \;=\; (0!, 1!, 2!, \ldots)$$

Acknowledgments

We are very much indebted to Ralf Hinze, who in [1, 2] gave a proof of Moessner's theorem in a calculation style. On hearing the presentation of our [4], he invited us to investigate a possible link with coinduction. The outcome of our investigation is the present proof.

References

[1] R. Hinze. Scans and convolutions - a calculational proof of Moessner's theorem. In *Proceedings of the 20th International Symposium on the Implementation and Application of Functional Languages (IFL '08)*, 2008.

[2] R. Hinze. Concrete stream calculus: an extended study. *Journal of Functional Programming*, pages 1–70, 2011.

[3] A. Moessner. Eine Bemerkung über die Potenzen der natürlichen Zahlen. *Aus den Sitzungsberichten der Bayerische Akademie der Wissenschaften, Mathematisch-naturwissenschaftliche Klasse 1951 Nr.3*, 1951.

[4] M. Niqui and J. Rutten. Sampling, splitting and merging in coinductive stream calculus. In C. Bolduc, J. Desharnais, and B. Ktari, editors, *MPC*, volume 6120 of *Lecture Notes in Computer Science*, pages 310–330. Springer, 2010. ISBN 978-3-642-13320-6.

[5] I. Paasche. Ein neuer Beweis des Moessnerschen Satz. *Aus den Sitzungsberichten der Bayerische Akademie der Wissenschaften, Mathematisch-naturwissenschaftliche Klasse 1952 Nr.1*, 1952.

[6] O. Perron. Beweis des Moessnerschen Satz. *Aus den Sitzungsberichten der Bayerische Akademie der Wissenschaften, Mathematisch-naturwissenschaftliche Klasse 1951 Nr.4*, 1951.

[7] J. Rutten. Universal coalgebra: a theory of systems. *Theoretical Computer Science*, 249(1):3–80, 2000. Fundamental Study.

[8] J. Rutten. A coinductive calculus of streams. *Mathematical Structures in Computer Science*, 15:93–147, 2005.

[9] H. Salié. Bemerkung zum einen Satz von Moessner. *Aus den Sitzungsberichten der Bayerische Akademie der Wissenschaften, Mathematisch-naturwissenschaftliche Klasse 1952 Nr.2*, 1952.