Acta Informatica (2020) 57:727–758
https://doi.org/10.1007/s00236-019-00353-7

ORIGINAL ARTICLE

Transducer degrees: atoms, infima and suprema

Jörg Endrullis[1] · Jan Willem Klop[1,2] · Rena Bakhshi[3]

Received: 12 April 2019 / Accepted: 14 November 2019 / Published online: 5 December 2019

Abstract
Although finite state transducers are very natural and simple devices, surprisingly little is known about the transducibility relation they induce on streams (infinite words). We collect some intriguing problems that have been unsolved since several years. The transducibility relation arising from finite state transduction induces a partial order of stream degrees, which we call *Transducer degrees*, analogous to the well-known *Turing degrees* or *degrees of unsolvability*. We show that there are pairs of degrees without supremum and without infimum. The former result is somewhat surprising since every finite set of degrees has a supremum if we strengthen the machine model to Turing machines, but also if we weaken it to Mealy machines.

1 Introduction

This paper is dedicated to Rob van Glabbeek, in the hope that the problems outlined in our paper evoke his interest, hopefully with the result that he applies his widely known problem solving powers to them. The open problems reported here have withstood several attempts to solve them. The problems arise in a very natural setting, centered around finite state transducers, devices that are natural in applications dealing with transforming infinite streams of symbols. In fact, we notice that in the general theory of automata we come across a relative *terra incognita* as soon as we come to automata as transducers on infinite words. Let us elaborate on this state of the art in automata theory.

Dedicated to Rob van Glabbeek in celebration of his 60th anniversary.

The first and third author have fond memories to their stay in Australia at NICTA, and the contacts there with Rob in 2013. The second author has equally fond memories to the years around 1985–1990 at CWI Amsterdam, when Rob was completing his groundbreaking Ph.D. thesis about comparative concurrency semantics.

✉ Rena Bakhshi
r.bakhshi@esciencecenter.nl

[1] Department of Computer Science, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands

[2] Centrum Wiskunde & Informatica (CWI), Amsterdam, The Netherlands

[3] The Netherlands eScience Center, Amsterdam, The Netherlands

Automata can be used as *acceptors* and *transducers*. As acceptors they simply accept or reject input words, and thereby define a language of accepted words. Transducers have a richer output. They transform input words into output words, and thereby realise a function on words.

Both aspects have been studied extensively for automata on *finite words*. There is also a large body of research on automata on infinite words, *streams* for short. Streams defined by finite automata, known as *automatic sequences*, play an important role in *number theory*. Automata for defining languages of streams are known as *stream automata* or *ω-automata*. There are various versions of stream automata, in particular *Büchi*, *Muller*, *Rabin* and *parity automata*. They are the foundation of *model checking* and *formal verification* as they allow for describing the (un)acceptable behaviours of non-terminating systems such as operating systems, control systems or hardware.

Surprisingly, finite automata for transforming streams have hardly been studied. An exception are Turing machines. The stream transformation realised by Turing machines gives rise to a pre-order on the set of streams: here, for streams u, w, we have $u \geq w$ if u can be transformed into w by some Turing machine. The ensuing hierarchy of stream degrees has been extensively studied and is known as *Turing degrees* or *degrees of unsolvability*. Beyond Turing machines, there has been almost no research on the power of finite automata, such as *finite state transducers* or *Mealy machines*, for transforming streams.

In this paper, we are interested in the hierarchy of stream degrees arising from finite state transduction. Again, we are interested in the pre-order $\geq$ on streams where $u \geq w$ if u can be transformed into w by some finite state transducer. Two streams u, w are considered equivalent, denoted $u \equiv w$, if $u \geq w$ and $w \geq u$. The equivalence classes of $\equiv$ are called *degrees*, and $\geq$ induces a partial order on these degrees (which we also denote by $\geq$). We refer to this hierarchy as the *Transducer degrees*. We are interested in the structural properties of this partial order; basic features concern the existence of:

(i) *bottom degree*, the set of ultimately periodic streams

$$\{uvvv\cdots \mid u, v \text{ finite words, } v \text{ non-empty}\};$$

(ii) *top degree*, not existing (however, the sub-hierarchy of computable degrees has a top-degree);
(iii) infinite *chains*, ascending and descending;
(iv) *atoms*, that is, minimal non-bottom degrees;
(v) *infima* and *suprema*, these are the greatest lower and least upper bounds;

Some of these questions have been addressed in previous work; see further [23]. In this paper we are interested in the existence of infima and suprema. Clearly this is analogous to the existence of greatest common divisor and least common divisor on the natural numbers. In this analogy, our atoms are the prime numbers. The present situation is different from the one of natural numbers in that suprema and infima now do not always exist. Indeed, we will construct, in a single joint construction, a pair of degrees that has no infimum and a pair of degrees that has no supremum. This result was announced in a preliminary way, still without proof, in [23].

1.1 Outline and contribution

In Sect. 2, we introduce finite state transducers and state some challenging questions that already arise at this stage. In Sect. 3, we introduce the partial order of *Transducer degrees* arising from the stream transformation realised by finite state transducers, and we describe the results of an investigation [12,17,19,20] of the appearance of atoms in the hierarchy of degrees. In Sects. 4 and 5, we introduce *mass products* as computational tools to reason about finite state transduction of *spiralling streams*. In Sect. 6, we characterise the transducts of spiralling streams in terms of displaced mass products.

In Sect. 7, we answer a question from [17] by showing that there are pairs of degrees without infimum and without supremum. The existence of such pairs of degrees without supremum is somewhat surprising since every finite set of degrees has a supremum if we strengthen the transformations to Turing machines (yielding Turing degrees), but also if we weaken it to Mealy machines (yielding Mealy degrees). In both cases, a supremum of a pair of degrees $\{u^{\equiv}, w^{\equiv}\}$ can be obtained by taking the degree of the stream of pairs $(u(0), w(0))\ ;\ (u(1), w(1))\ ; \cdots$

1.2 Related work

This paper is concerned with stream transformation via finite state transducers. We refer to the hierarchies of stream degrees arising from finite state transducers, Turing machines and Mealy machines as *Transducer degrees*, *Turing degrees* and *Mealy degrees*, respectively.

The Turing degrees have been the subject of extensive research, see for instance [30,33,34]. In contrast, Mealy degrees and Transducer degrees have hardly been studied. The partial order of Mealy degrees is studied by G. Rayna [31] and A. Belov [2]. The Transducer degrees have been studied in [12,17,19,20,23]. The papers [12,17,19,20] are concerned with atoms in this hierarchy. A comparison of Transducer degrees with the Turing degrees can be found in [23]. An interesting result about finite state transduction is due to M. Dekking [6] who has shown that every finite state transduct of a morphic stream is again morphic (or finite). Thus morphic sequences form a subhierarchy of the Transducer degrees.

Bosma and Zantema [5] study a hierarchy of two-sided infinite sequences arising from the transformation realised by permutation transducers. While the ultimately periodic sequences form the bottom degree under ordinary transduction, they split into three unpointed and seven pointed equivalence classes under permutation transduction. This study is continued in [39] with the focus on one-side infinite sequences.

2 Finite state transducers

Let us start with describing finite state transducers and give some examples how they operate on infinite streams. Already at this initial stage there are some easily stated but hard open problems.

A *finite state transducer (FST)* is a deterministic finite automaton which reads the input stream letter by letter, in each step producing an output word and changing its state. An example of an FST is depicted in Fig. 1.

We write ‘$a|w$’ along the transitions to indicate that the input letter is a and the output word is w. In a finite state transducer, the output word given by a transition can have arbitrary length (also empty).[1] The total output word is the concatenation of all the output words encountered along the edges.

[1] Thereby FSTs generalise the class of *Mealy machines*. The latter are restricted to output precisely one letter in each step. The transducer shown in Fig. 1 is not a Mealy machine, and there exists no Mealy machine implementing this transformation.

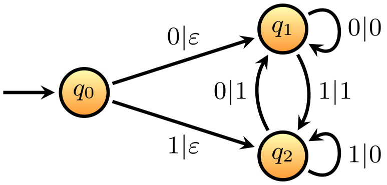


Fig. 1 A finite state transducer realizing the difference of consecutive bits modulo 2

Example 1 The transducer in Fig. 1 computes the difference of consecutive bits modulo 2. It transforms (*transduces*) the Thue–Morse sequence M into the period doubling sequence PD:

$$\begin{array}{cccccccccc} & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & \cdots = \mathsf{M} \\ \to & & 1 & 0 & 1 & 1 & 1 & 0 & 1 & \cdots = \mathsf{PD} \end{array}$$

The Thue–Morse sequence M is the fixed point of the morphism

$$\{0 \mapsto 01,\ 1 \mapsto 10\}$$

starting in 0. It can be obtained as the limit of iterating this morphism on the starting word 0:

$$\begin{aligned} & 0 \\ \mapsto\ & 01 \\ \mapsto\ & 0110 \\ \mapsto\ & 01101001 \\ & \vdots \end{aligned}$$

Likewise, the period doubling sequence PD can be obtained as the limit of iterating the morphism $\{0 \mapsto 11,\ 1 \mapsto 10\}$ on the starting word 1.

Formally, finite state transducers are defined as follows. (For a thorough introduction to finite state transducers, we refer to [1,32].) We write ε for the empty word and $\Sigma^\infty = \Sigma^* \cup \Sigma^{\mathbb{N}}$ for the set of finite and infinite words. We only consider *sequential* transducers and will simply speak of finite state transducers.

Definition 2 A *finite state transducer* $A = \langle \Sigma, \Delta, Q, q_0, \delta, \lambda \rangle$ consists of

(i) a finite input alphabet Σ,
(ii) a finite output alphabet Δ,
(iii) a finite set of states Q,
(iv) an initial state $q_0 \in Q$,
(v) a transition function $\delta : Q \times \Sigma \to Q$, and
(vi) an output function $\lambda : Q \times \Sigma \to \Delta^*$.

Whenever Σ and Δ are clear from the context, we write $A = \langle Q, q_0, \delta, \lambda \rangle$.

We extend the output and transition functions of the transducer from single letters to finite and infinite input words as follows.

Definition 3 Let $A = \langle \Sigma, \Delta, Q, q_0, \delta, \lambda \rangle$ be a finite state transducer. We extend the transition function δ from $Q \times \Sigma \to Q$ to $Q \times \Sigma^* \to Q$ by

$$\delta(q, \varepsilon) = q \qquad \delta(q, aw) = \delta(\delta(q, a), w) \qquad\qquad (q \in Q, a \in \Sigma, w \in \Sigma^*)$$

The output function λ is extended to $Q \times \Sigma^\infty \to \Delta^\infty$ by

$$\lambda(q, \varepsilon) = \varepsilon \qquad \lambda(q, aw) = \lambda(q, a) \cdot \lambda(\delta(q, a), w) \qquad\qquad (q \in Q, a \in \Sigma, w \in \Sigma^\infty)$$

We sometimes write $A(u)$ as shorthand for $\lambda(q_0, u)$. So the transducer A realises a function $A : \Sigma^{\mathbb{N}} \to \Delta^\infty$.

2.1 Challenging questions

Although finite state transducers are a very simple and elegant form of automata, hardly anything is known about their power in transforming streams. Even for simple examples of streams, there exist no techniques to determine whether one can be transformed into the other. To wit, consider the following problems:

Open Problem 1 *Consider the period doubling sequence* PD *and drop every third element:*

$$\begin{aligned}\mathsf{PD} &= 1011\ 1010\ 1011\ 1011\ 1011 \cdots \\ \mathsf{PD}' &= 10\_1\ 1\_10\ \_01\_\ 10\_1\ 1\_11 \cdots\end{aligned}$$

It is easy to find a finite state transducer that transforms PD *into* PD′*. Is the reverse also possible, or is information irrevocably lost?*

Open Problem 2 *The Mephisto waltz sequence*

$$\mathsf{W} = 001001110001001110110110001 \cdots$$

is obtained as the limit of iterating the morphism $\{0 \mapsto 001,\ 1 \mapsto 110\}$ *on the starting word* 0. *Can the Thue–Morse sequence* M *be transformed into* W *via a finite state transducer?*[2] *Is the reverse possible?*

There are currently no techniques available to answer such simple questions. We will come back these problems in the next section.

3 Transducer degrees

The transducibility relation arising from finite state transduction induces a partial order of stream degrees, which we call *Transducer degrees*. Here a stream u is at least as complex as w if u can be transformed into w by a finite state transducer. If the reverse is also possible, then both streams have the same complexity. A *degree* is an equivalence class of streams that have the same complexity.

The Transducer degrees are analogous to—but much more fine-grained than—the recursion-theoretic *degrees of unsolvability* or *Turing degrees*. Turing degrees have been the subject of extensive research in the 60's and 70's of the last century with many fascinating results and techniques (see for instance [27,30,33–36]). In the Turing degrees, sets of natural numbers are compared by means of transducibility using Turing machines. Note that a set of natural numbers is also a stream over the alphabet {0, 1} via its characteristic function. Thus the degrees of unsolvability can equivalently be considered as a hierarchy of stream degrees. Then we have Turing machines transforming streams into each other.

[2] If such a transducer exists, then it must be an *erasing* transducer. That is, at least one of the output words along the edges must be empty. A non-erasing transducer cannot do the transformation since it preserves α-substitutivity [37].

For a complexity comparison, Turing machines are too strong. We are typically interested in computable streams, but they are all identified by transducibility via Turing machines. In the hierarchy of Turing degrees, all computable streams are trivialised in the bottom degree. We are therefore interested in studying transducibility of streams with respect to less powerful devices, such as finite state transducers. A reduction of the computational power results in a finer structure of degrees. This point of view is also pursued in [39], further restricting to permutation transducers, leading to an even more refined hierarchy.

Here is the formal definition of the partial order of Transducer degrees.

Definition 4 Let Σ, Γ be finite alphabets, and $u \in \Sigma^{\mathbb{N}}$, $w \in \Gamma^{\mathbb{N}}$ streams. Let $A = \langle \Sigma, \Delta, Q, q_0, \delta, \lambda \rangle$ be a FST. We write $u \geq_A w$ if $w = \lambda(q_0, u)$. We write

$$u \geq w \ ,$$

and say that w is a *transduct* of u, if there exists a FST A such that $u \geq_A w$.

We write $u \equiv w$ if a forth and a back transformation is possible, that is, $u \geq w$ and $w \geq u$. Thus,

$$\equiv \ = \ (\geq \cap \leq).$$

We say that streams related by $\equiv$ are *equivalent*.

It is easily checked that $\equiv$ forms an equivalence relation, and we refer to the equivalence classes of $\equiv$ as *degrees*.

Every stream over a finite alphabet is equivalent to some stream over $\{0, 1\}$. So, every degree contains a representative from the set $\{0, 1\}^{\mathbb{N}}$. Thus, when investigating the partial order of stream degrees, it suffices to consider streams over the alphabet $\mathbf{2} = \{0, 1\}$.

Definition 5 The *degree* $u^{\equiv}$ of a stream $u \in \mathbf{2}^{\mathbb{N}}$ is the equivalence class of u with respect to $\equiv$, that is:

$$u^{\equiv} = \{w \in \mathbf{2}^{\mathbb{N}} \mid u \equiv w\}.$$

We write $\mathbf{2}^{\mathbb{N}}/_{\equiv}$ to denote the set of degrees $\{u^{\equiv} \mid u \in \mathbf{2}^{\mathbb{N}}\}$.

The transducibility relation $\geq$ induces a partial order on the set of degrees $\mathbf{2}^{\mathbb{N}}/_{\equiv}$. We refer to this partial order as *Transducer degrees*.

Definition 6 The *Transducer degrees* are the partial order $\langle \mathbf{2}^{\mathbb{N}}/_{\equiv}, \ \geq \ \rangle$ where

$$u^{\equiv} \geq w^{\equiv} \quad \Longleftrightarrow \quad u \geq w$$

for all words $u, w \in \mathbf{2}^{\mathbb{N}}$.

Figure 2 displays some initial results about the hierarchy of Transducer degrees that have been obtained in [12,17,19,20,23], and points to some open problems. We use the following notation: for $f : \mathbb{N} \to \mathbb{N}$ we define the stream

$$\langle f \rangle = \prod_{i=0}^{\infty} 10^{f(i)} = 10^{f(0)}\, 10^{f(1)}\, 10^{f(2)} \ \cdots \ .$$

We say that f is the *block function* of $\langle f \rangle$. By a *block* we refer to an occurrence of a factor in $\langle f \rangle$ of the shape $1\,\underbrace{00\cdots0}_{\text{0 or more}}$ that is followed by a 1.

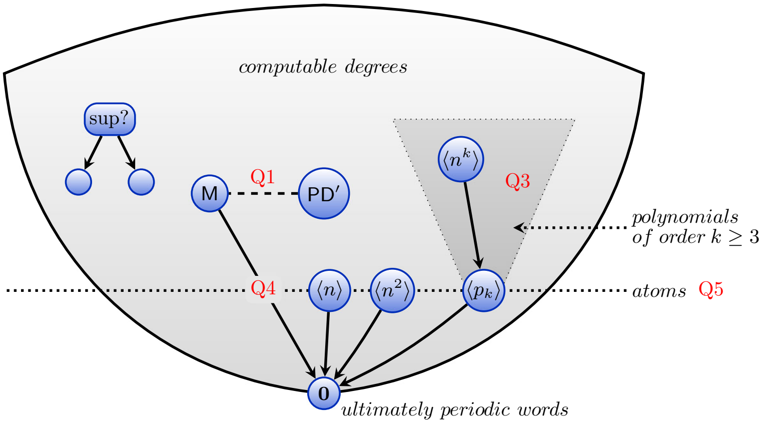


Fig. 2 The partial order of Transducer degrees. Some open problems are indicated in red: 1, …, 5 refer to open problems 1, …, 5, respectively. The blue nodes are degrees. Note that $\langle n \rangle$ and $\langle n^2 \rangle$ are atoms, while $\langle n^k \rangle$ is not an atom for $k \geq 3$. The notation $\langle \cdot \rangle$ is defined above. Here p_k is a particular polynomial of order k. The degree of $\langle p_k \rangle$ is an atom and all other polynomials of order k can be transduced to $\langle p_k \rangle$, but not vice versa (color figure online)

We often write $\langle f(n) \rangle$ to denote the sequence $\langle n \mapsto f(n) \rangle$. For instance:

$$\langle n \rangle = 1\ 10\ 100\ 1000\ 10000\ 100000\ \cdots$$
$$\langle n^2 \rangle = 1\ 10\ 10000\ 1000000000\ \cdots$$

The stream $\langle n \rangle$ is called 'rarified ones' in [26].

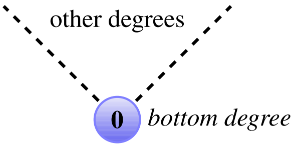
To get a better understanding of the hierarchies as mentioned, we discuss a few basic properties: bottom degrees and atoms.

3.1 Initial observations

An initial study of this partial order of degrees has been carried out in [17]. The hierarchy (displayed in Fig. 2) is not dense, not well-founded, there exist no maximal degrees, and a set of degrees has an upper bound if and only if the set is countable. A comparison with the well-known Turing degrees can be found in [23].

3.2 Bottom degree

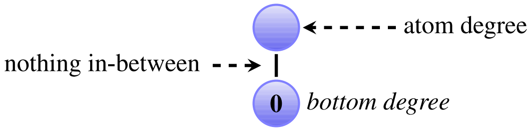
The *bottom degree* $\mathbf{0}$ is a degree that is less than or equal to all other degrees:



That is, for all degrees $\mathbf{x}$ we have $\mathbf{0} \leq \mathbf{x}$. For the Turing degrees the bottom degree consists of all computable streams. For the Transducer degrees, it consists of all *ultimately periodic streams*, streams of the form $uvvvv\cdots$ for finite words u, v.

3.3 Atoms

An interesting concept is that of an *atom degree*, that is, a degree that is directly above the bottom degree with nothing in-between:



Thus the atom degrees reduce only to **0** or themselves. Intuitively, the information content of a stream residing in an atom degree is 'indivisible': whatever transducer is applied on this stream, either the result is ultimately periodic (the structure is entirely destroyed), or there is enough structure left for a transducer to reconstruct the original stream.

(i) In the Mealy degrees, there exist no atom degrees, see further [2].
(ii) In the Turing degrees, atoms are usually referred to as *minimal degrees*. A famous result about Turing degrees, obtained by Spector [36], is the existence of an atom degree strictly below the first Turing jump (the degree of the halting problem). Lacombe [33] has extended the construction of Spector to show that there are continuum many atoms in the Turing degrees. So, embedded in the Transducer degrees, these streams would be uncomputable; hence above our focus of interest.

For the Transducer degrees, the existence of atoms has been investigated in the papers [12, 17,19,20]. We give a short account of our findings there. To discern atoms we focused on a substructure of the degrees hierarchy inhabited by streams with polynomial block function. We found that

(i) the degree of $\langle n \rangle$ is an atom,
(ii) the degree of $\langle n^2 \rangle$ is an atom, and
(iii) the degree of $\langle n^k \rangle$ is *not* an atom for $k \geq 3$.

Moreover, we found that for every $k \geq 1$, there is a unique atom among the degrees of polynomials of order k, namely the degree of $\langle p_k(n) \rangle$ where

$$p_k(n) = \sum_{i=0}^{k-1} (kn + i)^k .$$

It turns out that for every polynomial $q(n)$ of order k, we have $\langle q(n) \rangle \geq \langle p(n) \rangle$. So, the degree of $\langle p(n) \rangle$ is the infimum of all degrees of polynomials of order k. This is illustrated in Fig. 3.

So, the p_k is somehow "generic". We have

$$\begin{aligned} p_1(n) &= n \\ p_2(n) &= (2n+0)^2 + (2n+1)^2 = 8n^2 + 4n + 1 \\ &\vdots \end{aligned}$$

Open Problem 3 *How many degrees exist among polynomials of order k? What is the structure of the degrees of polynomials? In particular, is there a degree between $\langle n^k \rangle^{\equiv}$ and $\langle p_k(n) \rangle^{\equiv}$, for $k \geq 3$?*

Summarising, there is at least a countably infinite number of atom degrees. But it remains an open problem whether there are continuum many.

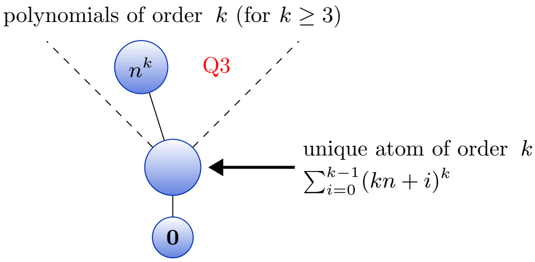

polynomials of order k (for $k \geq 3$)
n^k
Q3
unique atom of order k
$\sum_{i=0}^{k-1}(kn+i)^k$
0

Fig. 3 Structure of degrees of polynomials of order $k \geq 3$. The node 3 points to open problem 3 concerning the view on this part of the degree structure

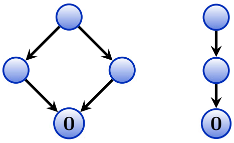


Fig. 4 Are these structures possible in the Transducer degrees?

3.4 Further problems for transducer degrees

Beyond these initial observations, the structure of the Transducer degrees is largely unexplored territory. There is a plethora of interesting further questions, including:

(i) Does every degree have a minimal cover, that is, a degree directly above with nothing in-between?
(ii) When does a pair of degrees have a least upper (greatest lower) bound?
(iii) Is every degree $\mathfrak{a}$ the greatest lower bound of a pair of degrees ($\neq \mathfrak{a}$)?
(iv) Are there interesting dense substructures? Are there dense intervals? That is degrees $\mathfrak{a}$ and $\mathfrak{e}$ with $\mathfrak{a} \lneq \mathfrak{e}$ such that for all $\mathfrak{b}, \mathfrak{d}$ with $\mathfrak{a} \leq \mathfrak{b} \lneq \mathfrak{d} \leq \mathfrak{e}$, there exists $\mathfrak{c}$ with $\mathfrak{b} \lneq \mathfrak{c} \lneq \mathfrak{d}$.
(v) How complex is the first-order theory in the language $\langle \geq, = \rangle$?
(vi) Are there suitable notions of Kolmogorov complexity that have relations to the Transducer degrees.

Further, we want to highlight the following open questions.

Open Problem 4 *Is the degree of Thue-Morse an atom?*

Open Problem 5 *Are there continuum many atoms? Does there exist some non-computable atom? Here a degree is* non-computable *if it contains a non-computable stream (then all streams in the degree are non-computable).*

Open Problem 6 *Can every finite partial order be embedded in the hierarchy? Can every finite distributive lattice be embedded in the hierarchy? In particular:*

(i) *Is there a degree that has precisely* three *degrees below itself: two incomparable degrees and the bottom degree?*
(ii) *Is there a degree that has precisely* two *degrees below itself?*

Structures (*i*) *and* (*ii*) *are displayed in Fig. 4 on the left and right, respectively.*

4 The algebra of masses

To introduce our counterexample (to the existence of suprema and infima in the Transducer degrees) in the next section, we need some preparations. In particular, we need certain linear operations on functions $f : \mathbb{N} \to \mathbb{Q}$ which we refer to as *mass products*. Many of these tools are from [12,19,20]. However, we simplify the presentation and strengthen some of the results.

4.1 Mass products and displacements

Definition 7 A *weight* is a non-empty tuple of non-negative rational numbers. A weight $\langle a_0, \ldots, a_{k-1} \rangle$ is called *zero* if $a_0, \ldots, a_{k-1} = 0$.

Let $\boldsymbol{\alpha} = \langle \alpha_0, \ldots, \alpha_{\ell-1} \rangle$ and $\boldsymbol{\beta} = \langle \beta_0, \ldots, \beta_{\ell'-1} \rangle$ be tuples. We define

- the *length* $|\boldsymbol{\alpha}| = \ell$,
- the *rotation* $\boldsymbol{\alpha}^{(0)} = \boldsymbol{\alpha}$ and $\boldsymbol{\alpha}^{(n+1)} = \langle \alpha_1, \ldots, \alpha_{\ell-1}, \alpha_0 \rangle^{(n)}$,
- the *concatenation* $\boldsymbol{\alpha} \,;\, \boldsymbol{\beta} = \langle \alpha_0, \ldots, \alpha_{\ell-1}, \beta_0, \ldots, \beta_{\ell'-1} \rangle$, and
- the *unfolding* $\boldsymbol{\alpha}^n$ for $n > 0$ by induction: $\boldsymbol{\alpha}^1 = \boldsymbol{\alpha}$ and $\boldsymbol{\alpha}^{n+1} = \boldsymbol{\alpha} \,;\, \boldsymbol{\alpha}^n$.

Definition 8 A *mass* is a non-empty tuple of weights. A *positive* mass consists only of non-zero weights.

For a mass $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \ldots, \boldsymbol{\alpha_{\ell-1}} \rangle$, we define

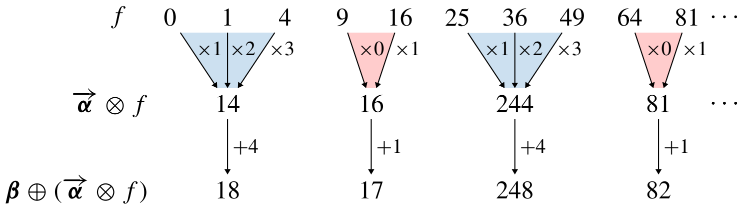
$$||\overrightarrow{\boldsymbol{\alpha}}|| = |\boldsymbol{\alpha_0}| + |\boldsymbol{\alpha_1}| + \cdots + |\boldsymbol{\alpha_{\ell-1}}|,$$

that is, the sum of the lengths of its entries.

Let A be a set and $f : \mathbb{N} \to A$ a function. We write $\mathcal{S}^k(f)$ for the *k-th shift* of f, defined by $\mathcal{S}^k(f)(n) = f(n+k)$. So, $\mathcal{S}^k(\,\cdot\,)$ is the operator removing the first k elements of a stream.

Our goal is to realise periodic transformations on streams as in Example 9. Here 'mass products' are a general format to transform streams by multiplying and grouping consecutive elements together, and 'displacements' are used for adding constants to the elements of a stream in a periodic fashion. Mass products and displacements are best understood by examples.

Example 9 Define $f : \mathbb{N} \to \mathbb{Q}$ by $f(n) = n^2$ for every $n \in \mathbb{N}$. Let $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \boldsymbol{\alpha_1} \rangle$ be a mass where $\boldsymbol{\alpha_0} = \langle 1, 2, 3 \rangle$ and $\boldsymbol{\alpha_1} = \langle 0, 1 \rangle$. Moreover, let $\boldsymbol{\beta} = \langle 4, 1 \rangle$. Then $\boldsymbol{\beta} \oplus (\overrightarrow{\boldsymbol{\alpha}} \otimes f)$ can be visualised as



f
0
1
4
9
16
25
36
49
64
81 ⋯
×1
×2
×3
×0
×1
×1
×2
×3
×0
×1
$\overrightarrow{\boldsymbol{\alpha}} \otimes f$
14
16
244
81
⋯
+4
+1
+4
+1
$\boldsymbol{\beta} \oplus (\overrightarrow{\boldsymbol{\alpha}} \otimes f)$
18
17
248
82

The weight $\boldsymbol{\alpha_0} = \langle 1, 2, 3 \rangle$ means that 3 consecutive entries are added while being multiplied by 1, 2 and 3, respectively.

A finite state transducer that realises the displaced mass product $\boldsymbol{\beta} \oplus (\overrightarrow{\boldsymbol{\alpha}} \otimes f)$ on the corresponding block streams is shown in Fig. 5. For every $f : \mathbb{N} \to \mathbb{N}$, the transducer transforms the block stream $\langle f \rangle$ to $\langle \boldsymbol{\beta} \oplus (\overrightarrow{\boldsymbol{\alpha}} \otimes f) \rangle$.

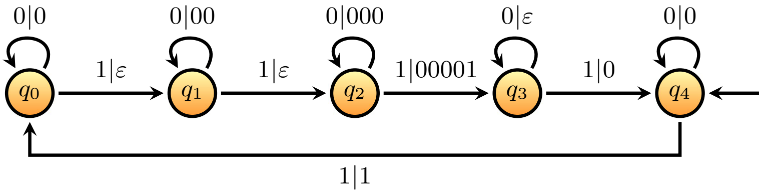


Fig. 5 A finite state transducer that realises the mass product and displacement given in Example 9 on the corresponding block stream

The implementation of these transformations is entirely straightforward, the details are as in the following definition.

Definition 10 Let $f : \mathbb{N} \to \mathbb{Q}$ be a function, $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \boldsymbol{\alpha_1}, \ldots, \boldsymbol{\alpha_{m-1}} \rangle$ a mass, $\boldsymbol{\alpha_0} = \langle a_0, \ldots, a_{k-1} \rangle$ a weight, and $\boldsymbol{\beta} = \langle b_0, b_1, \ldots, b_{m-1} \rangle \in \mathbb{Q}^+$.

(i) The *scalar product* $\boldsymbol{\alpha_0} \cdot f \in \mathbb{Q}$ is defined by

$$\boldsymbol{\alpha_0} \cdot f \;=\; a_0 f(0) + a_1 f(1) + \cdots + a_{k-1} f(k-1)\,.$$

(ii) The *mass product* $\overrightarrow{\boldsymbol{\alpha}} \otimes f : \mathbb{N} \to \mathbb{Q}$ is defined by induction on n:

$$\begin{aligned} (\overrightarrow{\boldsymbol{\alpha}} \otimes f)(0) &= \boldsymbol{\alpha_0} \cdot f \\ (\overrightarrow{\boldsymbol{\alpha}} \otimes f)(n+1) &= (\overrightarrow{\boldsymbol{\alpha}}^{(1)} \otimes \mathcal{S}^{|\boldsymbol{\alpha_0}|}(f))(n) \qquad (n \in \mathbb{N}) \end{aligned}$$

(iii) The *displacement* $\boldsymbol{\beta} \oplus f : \mathbb{N} \to \mathbb{Q}$ is defined by induction on n:

$$\begin{aligned} (\boldsymbol{\beta} \oplus f)(0) &= f(0) + b_0 \\ (\boldsymbol{\beta} \oplus f)(n+1) &= (\boldsymbol{\beta}^{(1)} \oplus \mathcal{S}^1(f))(n) \qquad (n \in \mathbb{N}) \end{aligned}$$

Remark 11 The transducer in Fig. 5 can also be viewed as an infinitary term rewrite system [18,24] operating on infinite words. The rewrite rules are:

$$\begin{array}{lllll} q_0 0 \to 0 q_0 & q_1 0 \to 00 q_1 & q_2 0 \to 000 q_2 & q_3 0 \to q_3 & q_4 0 \to 0 q_4 \\ q_0 1 \to q_1 & q_1 1 \to q_2 & q_2 1 \to 00001 q_3 & q_3 1 \to 0 q_4 & q_4 1 \to 1 q_0. \end{array}$$

Rewrite systems are more powerful than transducers. This flexibility can be convenient for reasoning about transformations of infinite words (or infinite terms). On the other hand, the increased power results in undecidability of many properties such as equivalence [9,15,16].

As a rewrite system, the properties termination (normalisation) and confluence are of interest; see for instance [7,21,22,28,38]. This rewrite system above is not infinitary normalising [11,29] on all terms due to the term $q_3 0000\cdots$ admitting an infinite divergent rewrite sequence. So, when viewing erasing transducers as rewrite systems, the notion of interest is that of *local* termination [8,25] and *local* productivity [13,14,40] on a given set of starting terms.

4.2 Computing with mass products

The following definitions help computing with mass products.

Definition 12 For a mass $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \ldots, \boldsymbol{\alpha_{\ell-1}} \rangle$ we define

$$\langle \alpha|_0, \alpha|_1, \ldots, \alpha|_{||\overrightarrow{\boldsymbol{\alpha}}||-1} \rangle = \boldsymbol{\alpha_0} \ ; \ \cdots \ ; \ \boldsymbol{\alpha_{\ell-1}}.$$

So, $\alpha|_i$ denotes the i-th element of the weight $\boldsymbol{\alpha_0} \ ; \ \cdots \ ; \ \boldsymbol{\alpha_{\ell-1}}$ (we count from 0).

Example 13 Let $\overrightarrow{\boldsymbol{\alpha}} = \langle \langle 3, 1, 5 \rangle, \langle 1 \rangle, \langle 2, 4 \rangle \rangle$. Then we have

$$\alpha|_0 = 3 \qquad \alpha|_1 = 1 \qquad \alpha|_2 = 5 \qquad \alpha|_3 = 1 \qquad \alpha|_4 = 2 \qquad \alpha|_5 = 4.$$

For a mass $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \ldots, \boldsymbol{\alpha_{\ell-1}} \rangle$, we define a function $\varphi(\overrightarrow{\boldsymbol{\alpha}}, \cdot)$ that maps the index i of an element $\alpha|_i$ to the number j of the weight $\boldsymbol{\alpha_j}$ that $\alpha|_i$ belongs to.

Definition 14 For a mass $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \ldots, \boldsymbol{\alpha_{\ell-1}} \rangle$, we define a function

$$\varphi(\overrightarrow{\boldsymbol{\alpha}}, \cdot) : \{0, \ldots, ||\overrightarrow{\boldsymbol{\alpha}}|| - 1\} \to \{0, \ldots, |\overrightarrow{\boldsymbol{\alpha}}| - 1\}$$

by

$$\varphi(\overrightarrow{\boldsymbol{\alpha}}, i) = \min\{k \in \{0, \ldots, \ell - 1\} \mid i < |\boldsymbol{\alpha_0}| + \cdots + |\boldsymbol{\alpha_k}|\}$$

for every $0 \leq i < ||\overrightarrow{\boldsymbol{\alpha}}||$.

Example 15 Let $\overrightarrow{\boldsymbol{\alpha}} = \langle \langle 3, 1, 5 \rangle, \langle 1 \rangle, \langle 2, 4 \rangle \rangle$. Then

$$\begin{array}{lll} \varphi(\overrightarrow{\boldsymbol{\alpha}}, 0) = 0 & \varphi(\overrightarrow{\boldsymbol{\alpha}}, 1) = 0 & \varphi(\overrightarrow{\boldsymbol{\alpha}}, 2) = 0 \\ \varphi(\overrightarrow{\boldsymbol{\alpha}}, 3) = 1 & \varphi(\overrightarrow{\boldsymbol{\alpha}}, 4) = 2 & \varphi(\overrightarrow{\boldsymbol{\alpha}}, 5) = 2. \end{array}$$

It is not hard to see that this amounts to simply stepping through a mass and recording the number of the weight you are in. So, $\overrightarrow{\boldsymbol{\alpha}}$ yields 000122 for the steps 012345.

The following lemma provides an alternative way to compute mass products.

Lemma 16 *Let* $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \ldots, \boldsymbol{\alpha_{\ell-1}} \rangle$ *be a mass. Then*

$$(\overrightarrow{\boldsymbol{\alpha}} \otimes f)(h|\overrightarrow{\boldsymbol{\alpha}}| + h') = \sum_{\substack{i \in \{0, \ldots, ||\overrightarrow{\boldsymbol{\alpha}}||-1\} \\ \varphi(\overrightarrow{\boldsymbol{\alpha}}, i) = h'}} \big(\alpha|_i \cdot f(h||\overrightarrow{\boldsymbol{\alpha}}|| + i) \big)$$

for every $f : \mathbb{N} \to \mathbb{Q}$, $h \in \mathbb{N}$ *and* $h' \in \{0, \ldots, |\overrightarrow{\boldsymbol{\alpha}}| - 1\}$. □

Example 17 We consider the mass product from Example 9. Let $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \boldsymbol{\alpha_1} \rangle$ where $\boldsymbol{\alpha_0} = \langle 1, 2, 3 \rangle$ and $\boldsymbol{\alpha_1} = \langle 0, 1 \rangle$. Then we have

$$\begin{array}{ccccc} \alpha|_0 = 1 & \alpha|_1 = 2 & \alpha|_2 = 3 & \alpha|_3 = 0 & \alpha|_4 = 1 \\ \varphi(\overrightarrow{\boldsymbol{\alpha}}, 0) = 0 & \varphi(\overrightarrow{\boldsymbol{\alpha}}, 1) = 0 & \varphi(\overrightarrow{\boldsymbol{\alpha}}, 2) = 0 & \varphi(\overrightarrow{\boldsymbol{\alpha}}, 3) = 1 & \varphi(\overrightarrow{\boldsymbol{\alpha}}, 4) = 1. \end{array}$$

and $|\overrightarrow{\boldsymbol{\alpha}}| = 2$, $||\overrightarrow{\boldsymbol{\alpha}}|| = 5$. We compute $(\overrightarrow{\boldsymbol{\alpha}} \otimes f)(3) = (\overrightarrow{\boldsymbol{\alpha}} \otimes f)(1|\overrightarrow{\boldsymbol{\alpha}}| + 1)$:

$$\begin{aligned} (\overrightarrow{\boldsymbol{\alpha}} \otimes f)(1|\overrightarrow{\boldsymbol{\alpha}}| + 1) &= \sum_{\substack{i \in \{0, \ldots, ||\overrightarrow{\boldsymbol{\alpha}}||-1\} \\ \varphi(\overrightarrow{\boldsymbol{\alpha}}, i) = 1}} \big(\alpha|_i \cdot f(1||\overrightarrow{\boldsymbol{\alpha}}|| + i) \big) \\ &= \alpha|_3 \cdot f(1||\overrightarrow{\boldsymbol{\alpha}}|| + 3) + \alpha|_4 \cdot f(1||\overrightarrow{\boldsymbol{\alpha}}|| + 4) \\ &= 0 \cdot 64 + 1 \cdot 81 = 81. \end{aligned}$$

4.3 Properties of mass products

Unfolding a mass does not change its semantics.

Lemma 18 *We have* $\overrightarrow{\boldsymbol{\alpha}} \otimes f = \overrightarrow{\boldsymbol{\alpha}}^{n} \otimes f$ *for every mass* $\overrightarrow{\boldsymbol{\alpha}}$, *function* $f : \mathbb{N} \to \mathbb{Q}$ *and natural number* $n > 0$. □

By definition of mass products we have

$$\mathcal{S}(\overrightarrow{\boldsymbol{\alpha}} \otimes f) = \overrightarrow{\boldsymbol{\alpha}}^{(1)} \otimes \mathcal{S}^{|\boldsymbol{\alpha_0}|}(f). \tag{1}$$

The following lemma generalises this to arbitrary shifts.

Lemma 19 *Let* $f : \mathbb{N} \to \mathbb{Q}$ *and* $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \dots, \boldsymbol{\alpha_{\ell-1}} \rangle$ *a mass. Then*

$$\mathcal{S}^{h|\overrightarrow{\boldsymbol{\alpha}}|+h'}(\overrightarrow{\boldsymbol{\alpha}} \otimes f) = \overrightarrow{\boldsymbol{\alpha}}^{(h')} \otimes \mathcal{S}^{h\|\overrightarrow{\boldsymbol{\alpha}}\|+\sum_{i=0}^{h'-1}|\boldsymbol{\alpha_i}|}(f)$$

for every $h \in \mathbb{N}$ *and* $h' \in \{0, \dots, |\overrightarrow{\boldsymbol{\alpha}}| - 1\}$.

Proof The claim is obvious for $h' = 0$. For $h' > 0$ it follows from (1) by induction on h'. □

4.4 Composition of mass products

We have already defined how a mass $\overrightarrow{\boldsymbol{\beta}}$ is applied to a function $f : \mathbb{N} \to \mathbb{Q}$ (a stream of rational numbers) with the result a function $\overrightarrow{\boldsymbol{\beta}} \otimes f$. Often we need to iterate such products, obtaining for a second mass $\overrightarrow{\boldsymbol{\alpha}}$ the product

$$\overrightarrow{\boldsymbol{\alpha}} \otimes (\overrightarrow{\boldsymbol{\beta}} \otimes f).$$

In analogy with ordinary function composition, satisfying $(f \circ g)(x) = f(g(x))$ for functions f, g, it will be convenient to have likewise a composition of masses, again denoted with $\otimes$, so that we have the 'associative rule'

$$\overrightarrow{\boldsymbol{\alpha}} \otimes (\overrightarrow{\boldsymbol{\beta}} \otimes f) = (\overrightarrow{\boldsymbol{\alpha}} \otimes \overrightarrow{\boldsymbol{\beta}}) \otimes f.$$

Note that the left-hand side is already defined, but the right-hand side not yet; we have to define $\overrightarrow{\boldsymbol{\alpha}} \otimes \overrightarrow{\boldsymbol{\beta}}$.

Bearing in mind how $\overrightarrow{\boldsymbol{\beta}} \otimes f$ is defined in Definition 10, it is straightforward how to proceed, in particular viewing the 'tripod' picture in Example 9. All we have to do is pipelining this tripod (in general 'n-pods' in periodic repetition) construction for $\overrightarrow{\boldsymbol{\beta}}$ with a second layer for $\overrightarrow{\boldsymbol{\alpha}}$ as in the following example and figure. Having this double layered tripod processing, we must then describe the effect of the two layers, for masses $\overrightarrow{\boldsymbol{\alpha}}$ and $\overrightarrow{\boldsymbol{\beta}}$, into a single layer, a mass called $\overrightarrow{\boldsymbol{\alpha}} \otimes \overrightarrow{\boldsymbol{\beta}}$. It is then clear by the construction that indeed $\overrightarrow{\boldsymbol{\alpha}} \otimes (\overrightarrow{\boldsymbol{\beta}} \otimes f) = (\overrightarrow{\boldsymbol{\alpha}} \otimes \overrightarrow{\boldsymbol{\beta}}) \otimes f$.

Of course, the extraction of the precise formal definition of $\overrightarrow{\boldsymbol{\alpha}} \otimes \overrightarrow{\boldsymbol{\beta}}$ requires meticulous care, but in principle is straightforward based on the periodical repetitions of the weights in both masses $\overrightarrow{\boldsymbol{\alpha}}$ and $\overrightarrow{\boldsymbol{\beta}}$. We give an example that displays the intuition, and provide the ensuing formal definition.

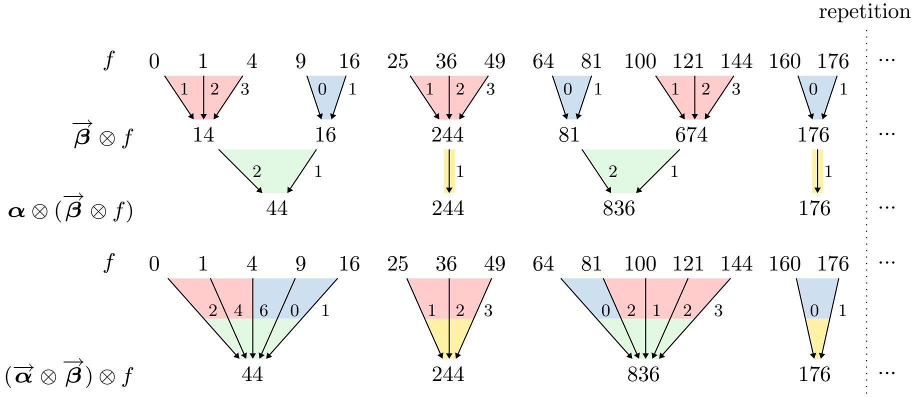


Fig. 6 The composition of masses derived from a two-layered 'tripod' construction

Example 20 Continuing Example 9, let $\vec{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \boldsymbol{\alpha_1} \rangle$ and $\vec{\boldsymbol{\beta}} = \langle \boldsymbol{\beta_0}, \boldsymbol{\beta_1} \rangle$, where

$$\boldsymbol{\alpha_0} = \langle 2, 1 \rangle \qquad \boldsymbol{\beta_0} = \langle 1, 2, 3 \rangle$$
$$\boldsymbol{\alpha_1} = \langle 1 \rangle \qquad \boldsymbol{\beta_1} = \langle 0, 1 \rangle .$$

The composition $\vec{\boldsymbol{\alpha}} \otimes \vec{\boldsymbol{\beta}}$ can be derived by the two-layered 'tripod' construction displayed in Fig. 6. We unfold (repeat) the masses $\vec{\boldsymbol{\alpha}}$ and $\vec{\boldsymbol{\beta}}$ such that we have $||\vec{\boldsymbol{\alpha}}|| = |\vec{\boldsymbol{\beta}}|$; then the tripods align and there is a repetition. Afterwards the edges are 'composed' by multiplying the labels along the path. We derive

$$\vec{\boldsymbol{\beta}} \otimes \vec{\boldsymbol{\alpha}} = \langle \langle 2, 4, 6,\ 0, 1 \rangle,\ \langle 1, 2, 3 \rangle,\ \langle 0, 2,\ 1, 2, 3 \rangle,\ \langle 0, 1 \rangle \rangle .$$

See Example 23 for the formal computation.

We now derive a formal definition of the composition $\vec{\boldsymbol{\alpha}} \otimes \vec{\boldsymbol{\beta}}$ which follows closely the idea of the construction displayed in Fig. 6.

Definition 21 For $a \in \mathbb{Q}_{\geq 0}$ and a weight $\boldsymbol{\alpha} = \langle \alpha_0, \ldots, \alpha_{\ell-1} \rangle$, we define:

$$a \cdot \boldsymbol{\alpha} = \langle a \cdot \alpha_0, \ldots, a \cdot \alpha_{\ell-1} \rangle \ .$$

If $\vec{\boldsymbol{\beta}} = \langle \boldsymbol{\beta_0}, \ldots, \boldsymbol{\beta_{\ell-1}} \rangle$ is a mass, we define the *scalar product*

$$\boldsymbol{\alpha} \cdot \vec{\boldsymbol{\beta}} = (\alpha_0 \cdot \boldsymbol{\beta_0}) \ ; \ \cdots \ ; \ (\alpha_{\ell-1} \cdot \boldsymbol{\beta_{\ell-1}}).$$

Definition 22 The *product* $\vec{\boldsymbol{\alpha}} \otimes \vec{\boldsymbol{\beta}}$ of masses $\vec{\boldsymbol{\alpha}}$ and $\vec{\boldsymbol{\beta}}$ is defined as follows. Let $\vec{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \ldots, \boldsymbol{\alpha_{\ell-1}} \rangle$ and $\vec{\boldsymbol{\beta}} = \langle \boldsymbol{\beta_0}, \ldots, \boldsymbol{\beta_{\ell'-1}} \rangle$.

If $||\vec{\boldsymbol{\alpha}}|| = |\vec{\boldsymbol{\beta}}|$, define

$$\vec{\boldsymbol{\alpha}} \otimes \vec{\boldsymbol{\beta}} = (\boldsymbol{\alpha_0} \cdot \vec{\boldsymbol{\gamma_0}}) \ ; \ (\boldsymbol{\alpha_1} \cdot \vec{\boldsymbol{\gamma_1}}) \ ; \ \cdots \ ; \ (\boldsymbol{\alpha_{\ell-1}} \cdot \overrightarrow{\boldsymbol{\gamma_{\ell-1}}}),$$

where, for $0 \leq i < \ell$,

$$\vec{\boldsymbol{\gamma_i}} = \langle \boldsymbol{\beta_{k_i}}, \boldsymbol{\beta_{k_i+1}}, \ldots, \boldsymbol{\beta_{k_i+|\alpha_i|-1}} \rangle$$
$$k_i = ||\langle \boldsymbol{\alpha_0}, \boldsymbol{\alpha_1}, \ldots, \boldsymbol{\alpha_{i-1}} \rangle|| = |\boldsymbol{\alpha_0}| + |\boldsymbol{\alpha_1}| + \cdots + |\boldsymbol{\alpha_{i-1}}|.$$

If $||\overrightarrow{\boldsymbol{\alpha}}|| \neq |\overrightarrow{\boldsymbol{\beta}}|$, define

$$\overrightarrow{\boldsymbol{\alpha}} \otimes \overrightarrow{\boldsymbol{\beta}} = \overrightarrow{\boldsymbol{\alpha}}^{c/||\overrightarrow{\boldsymbol{\alpha}}||} \otimes \overrightarrow{\boldsymbol{\beta}}^{c/|\overrightarrow{\boldsymbol{\beta}}|},$$

where c is the least common multiple of $||\overrightarrow{\boldsymbol{\alpha}}||$ and $|\overrightarrow{\boldsymbol{\beta}}|$.

Example 23 We continue Example 20. We compute $\overrightarrow{\boldsymbol{\alpha}} \otimes \overrightarrow{\boldsymbol{\beta}}$. We have $||\overrightarrow{\boldsymbol{\alpha}}|| = 3$ and $|\overrightarrow{\boldsymbol{\beta}}| = 2$. Thus, we have to unfold $\overrightarrow{\boldsymbol{\alpha}}$ twice and $\overrightarrow{\boldsymbol{\beta}}$ thrice:

$$\overrightarrow{\boldsymbol{\alpha}}^2 = \langle \boldsymbol{\alpha_0}, \boldsymbol{\alpha_1}, \boldsymbol{\alpha_0}, \boldsymbol{\alpha_1} \rangle \qquad \overrightarrow{\boldsymbol{\beta}}^3 = \langle \boldsymbol{\beta_0}, \boldsymbol{\beta_1}, \boldsymbol{\beta_0}, \boldsymbol{\beta_1}, \boldsymbol{\beta_0}, \boldsymbol{\beta_1} \rangle.$$

Then

$$\begin{aligned}
\overrightarrow{\boldsymbol{\alpha}} \otimes \overrightarrow{\boldsymbol{\beta}} &= \overrightarrow{\boldsymbol{\alpha}}^2 \otimes \overrightarrow{\boldsymbol{\beta}}^3 \\
&= \langle \boldsymbol{\alpha_0}, \boldsymbol{\alpha_1}, \boldsymbol{\alpha_0}, \boldsymbol{\alpha_1} \rangle \otimes \langle \boldsymbol{\beta_0}, \boldsymbol{\beta_1}, \boldsymbol{\beta_0}, \boldsymbol{\beta_1}, \boldsymbol{\beta_0}, \boldsymbol{\beta_1} \rangle \\
&= (\boldsymbol{\alpha_0} \cdot \langle \boldsymbol{\beta_0}, \boldsymbol{\beta_1} \rangle) \;;\; (\boldsymbol{\alpha_1} \cdot \langle \boldsymbol{\beta_0} \rangle) \;;\; (\boldsymbol{\alpha_0} \cdot \langle \boldsymbol{\beta_1}, \boldsymbol{\beta_0} \rangle) \;;\; (\boldsymbol{\alpha_1} \cdot \langle \boldsymbol{\beta_1} \rangle) \\
&= \langle \langle 2, 4, 6,\ 0, 1 \rangle,\ \langle 1, 2, 3 \rangle,\ \langle 0, 2,\ 1, 2, 3 \rangle,\ \langle 0, 1 \rangle \rangle.
\end{aligned}$$

Lemma 24 *We have $\overrightarrow{\boldsymbol{\alpha}} \otimes (\overrightarrow{\boldsymbol{\beta}} \otimes f) = (\overrightarrow{\boldsymbol{\alpha}} \otimes \overrightarrow{\boldsymbol{\beta}}) \otimes f$ for every function $f : \mathbb{N} \to \mathbb{Q}$, and all masses $\overrightarrow{\boldsymbol{\alpha}}$, $\overrightarrow{\boldsymbol{\beta}}$.*

Proof Left to the reader; see the construction in Fig. 6 and Example 20. □

4.5 Operations that leave the degree unchanged

Displacements leave the degree of the associated block sequence unchanged.

Lemma 25 *We have $\langle f \rangle \equiv \langle \boldsymbol{\beta} \oplus f \rangle$ for every $f : \mathbb{N} \to \mathbb{Q}_{\geq 0}$ and every $\boldsymbol{\beta} \in \mathbb{Q}^+$ such that $\boldsymbol{\beta} \oplus f$ is non-negative.*

Proof Finite state transducers can add (and remove) a constant amount of zeros to (from) blocks $10 \cdots 0$ in a periodic fashion. □

The next lemma enables us to 'drop' zero weights from masses, while leaving the degree of the corresponding block stream unchanged.

Lemma 26 *Let $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \ldots, \boldsymbol{\alpha_{\ell-1}} \rangle$ be a mass such that $\ell \geq 2$ and $\boldsymbol{\alpha_k}$ is zero for some $0 \leq k < \ell$.*

(i) If $k > 0$, define

$$\overrightarrow{\boldsymbol{\alpha}'} = \langle \boldsymbol{\alpha_0}, \ldots, \boldsymbol{\alpha_{k-2}}, \boldsymbol{\beta}, \boldsymbol{\alpha_{k+1}}, \ldots, \boldsymbol{\alpha_{\ell-1}} \rangle \qquad \boldsymbol{\beta} = \boldsymbol{\alpha_{k-1}} \;;\; \boldsymbol{\alpha_k}.$$

(ii) If $k < \ell - 1$, define

$$\overrightarrow{\boldsymbol{\alpha}'} = \langle \boldsymbol{\alpha_0}, \ldots, \boldsymbol{\alpha_{k-1}}, \boldsymbol{\beta}, \boldsymbol{\alpha_{k+2}}, \ldots, \boldsymbol{\alpha_{\ell-1}} \rangle \qquad \boldsymbol{\beta} = \boldsymbol{\alpha_k} \;;\; \boldsymbol{\alpha_{k+1}}.$$

In both cases, we have $\langle \overrightarrow{\boldsymbol{\alpha}} \otimes f \rangle \equiv \langle \overrightarrow{\boldsymbol{\alpha}'} \otimes f \rangle$ for every $f : \mathbb{N} \to \mathbb{N}$.

Proof Let $f : \mathbb{N} \to \mathbb{N}$. Let $g = \overrightarrow{\boldsymbol{\alpha}} \otimes f$ and $g' = \overrightarrow{\boldsymbol{\alpha'}} \otimes f$. Since $\boldsymbol{\alpha_k}$ is zero, we have that $g(k + \ell n) = 0$ for every $n \in \mathbb{N}$. We consider the cases (i) and (ii).

For case (i), the mass $\overrightarrow{\boldsymbol{\alpha'}}$ is obtained from $\overrightarrow{\boldsymbol{\alpha}}$ by merging (concatenating) the weights $\boldsymbol{\alpha_{k-1}}$ and $\boldsymbol{\alpha_k}$. Likewise, for case (ii), the mass $\overrightarrow{\boldsymbol{\alpha'}}$ is obtained from $\overrightarrow{\boldsymbol{\alpha}}$ by concatenating $\boldsymbol{\alpha_k}$ and $\boldsymbol{\alpha_{k+1}}$.

In both cases, thinking of g and g' as sequences of natural numbers, it follows that g' is obtained from g by dropping the elements at indices $k + \ell n$ for $n \in \mathbb{N}$. As we have $g(k + \ell n) = a_k$ for every $n \in \mathbb{N}$, it follows that $\langle g \rangle \equiv \langle g' \rangle$ since a finite state transducer can easily realise the periodic insertion or removal of blocks of size 0. Thus, $\langle \overrightarrow{\boldsymbol{\alpha}} \otimes f \rangle \equiv \langle \overrightarrow{\boldsymbol{\alpha'}} \otimes f \rangle$ in both cases. □

Example 27 To illustrate Lemma 26, let

$$\overrightarrow{\boldsymbol{\alpha}} = \langle \langle 1, 2 \rangle, \langle 0, 0 \rangle, \langle 5 \rangle, \langle 2, 3 \rangle \rangle.$$

Then, in the lemma, we have

(i) $\overrightarrow{\boldsymbol{\alpha'}} = \langle \langle 1, 2, 0, 0 \rangle, \langle 5 \rangle, \langle 2, 3 \rangle \rangle$, or
(ii) $\overrightarrow{\boldsymbol{\alpha'}} = \langle \langle 1, 2 \rangle, \langle 0, 0, 5 \rangle, \langle 2, 3 \rangle \rangle$.

In both cases, we have $\langle \overrightarrow{\boldsymbol{\alpha}} \otimes f \rangle \equiv \langle \overrightarrow{\boldsymbol{\alpha'}} \otimes f \rangle$ for every $f : \mathbb{N} \to \mathbb{N}$.

Theorem 28 *Let $\overrightarrow{\boldsymbol{\alpha}}$ be a mass containing at least one non-zero weight. Then there exists a positive mass $\overrightarrow{\boldsymbol{\alpha'}}$ such that*

$$\langle \overrightarrow{\boldsymbol{\alpha}} \otimes f \rangle \equiv \langle \overrightarrow{\boldsymbol{\alpha'}} \otimes f \rangle$$

for every $f : \mathbb{N} \to \mathbb{N}$.

Proof By repeated application of Lemma 26 we can drop all zero weights from α. This yields $\langle \overrightarrow{\boldsymbol{\alpha}} \otimes f \rangle \equiv \langle \overrightarrow{\boldsymbol{\alpha'}} \otimes f \rangle$ for mass $\overrightarrow{\boldsymbol{\alpha'}}$ consisting only of non-zero weights. □

5 The algebra of spiralling functions

For our construction, we consider the subhierarchy of degrees based on the interesting class of *spiralling* number-theoretic functions [3,4,12] and as a computational device, a form of scalar products.

Definition 29 A function $f : \mathbb{N} \to \mathbb{N}$ is called *spiralling* if

(i) $\lim_{n\to\infty} f(n) = \infty$, and
(ii) for every $m \geq 1$, the function $n \mapsto f(n) \bmod m$ is ultimately periodic.

A stream $\langle f \rangle$ is called *spiralling* whenever f is spiralling.

Spiralling functions are called ‘cyclically ultimately periodic’ in the literature, see for instance, [3,4].[3] The class of spiralling functions is closed under addition and multiplication. It is also closed under the addition of constants and the multiplication with non-zero

[3] The papers [3,4] consider regularity preserving operations on regular languages. There spiralling functions are employed in the context of regularity preserving filter operations. See also [10] for functions that preserve but do not reflect regularity.

constants. For instance, polynomials with natural numbers as coefficients are spiralling (if they are not constant).

Note that for every $f : \mathbb{N} \to \mathbb{N}$ and $d \in \mathbb{N}_{>0}$, $\langle f \rangle \equiv \langle d \cdot f \rangle$. So, we may ignore constant factors as they do not change the degree of the stream. This enables us to use functions $f : \mathbb{N} \to \mathbb{Q}_{\geq 0}$ with, as it were, 'rational block size', as long as there exists $d \in \mathbb{N}_{>0}$ such that $d \cdot f : \mathbb{N} \to \mathbb{N}$.

Definition 30 A function $f : \mathbb{N} \to \mathbb{Q}_{\geq 0}$ has a *common denominator* if there exists $d \in \mathbb{N}_{>0}$ such that $d \cdot f(n) \in \mathbb{N}$ for every $n \in \mathbb{N}$.

For a function $f : \mathbb{N} \to \mathbb{Q}_{\geq 0}$ with a common denominator, we define

$$[f] = d \cdot f \qquad\qquad \text{and} \qquad\qquad \langle f \rangle = \langle [f] \rangle,$$

where $d \in \mathbb{N}_{>0}$ is minimal with the property $d \cdot f(n) \in \mathbb{N}$ for every $n \in \mathbb{N}$.

Definition 31 A function $f : \mathbb{N} \to \mathbb{Q}_{\geq 0}$ is called *spiralling* if

(i) f has a common denominator, and
(ii) $[f]$ is spiralling.

We write 🌀 to denote the set of spiralling function from $\mathbb{N} \to \mathbb{Q}_{\geq 0}$. We use 🌀$_{\mathbb{N}}$ to denote the spiralling functions from $\mathbb{N} \to \mathbb{N}$.

Definition 32 A function $f : \mathbb{N} \to \mathbb{Q}$ is called *non-negative* if $f(n) \geq 0$ for every $n \in \mathbb{N}$.

The next two lemmas follow straightforwardly by the respective definitions and elementary arithmetic. The existence of a common denominator is preserved under displacements and mass products.

Lemma 33 *Let $f : \mathbb{N} \to \mathbb{Q}_{\geq 0}$ be a function with common denominator. Let $\overrightarrow{\boldsymbol{\alpha}}$ be a mass, and let $\boldsymbol{\beta} \in \mathbb{Q}^+$ such that $\boldsymbol{\beta} \oplus f$ is non-negative. Then the functions $\overrightarrow{\boldsymbol{\alpha}} \otimes f$ and $\boldsymbol{\beta} \oplus f$ have a common denominator.* □

The class of spiralling sequences is closed under shifts, displacements and mass products with positive masses.

Lemma 34 *Let $f \in$ 🌀. Then*

(i) $S^n(f) \in$ 🌀 *for every $n \in \mathbb{N}$,*

(ii) $\beta \oplus f \in$ 🌀 *for every $\boldsymbol{\beta} \in \mathbb{Q}^+$ such that $\boldsymbol{\beta} \oplus f$ is non-negative, and*

(iii) $\overrightarrow{\alpha} \otimes f \in$ 🌀 *for every positive mass $\overrightarrow{\boldsymbol{\alpha}}$.* □

Lemma 35 *For every positive integer p, and $f \in$ 🌀$_{\mathbb{N}}$, $(n \mapsto p^{f(n)}) \in$ 🌀$_{\mathbb{N}}$.*

Proof Let $m \in \mathbb{N}$. We show that $p^{f(n)}$ is ultimately periodic modulo m. By the pigeonhole principle, there are $a, b > 0$ such that $p^a \equiv p^{a+b} (\operatorname{mod} m)$. Hence, we have $(\star)$ $p^{a'} \equiv p^{a'+kb} (\operatorname{mod} m)$ for all $a' \geq a$ and $k \in \mathbb{N}$. Since f is ultimately periodic modulo b, there exist $n_0, r \in \mathbb{N}$ such that $f(n) \equiv f(n+r) (\operatorname{mod} b)$ for all $n \geq n_0$. Now let $N \geq n_0$ be such that $f(n) > a$ for all $n \geq N$. Then for all $n > N$, there exist $a' \geq a$ and $k_1, k_2 \in \mathbb{N}$ such that $f(n) = a' + k_1 b$ and $f(n+r) = a' + k_2 b$. Then by $(\star)$ we conclude $p^{f(n)} \equiv p^{a'} \equiv p^{f(n+r)} (\operatorname{mod} m)$. □

Clearly, the identity function is spiralling, hence, by Lemma 35, p^n is spiralling. A second application of the lemma yields that also p^{p^n} is a spiralling function.

Remark 36 The condition that $\lim_{n\to\infty} f(n)$ exists in the definition of spiralling is necessary for Lemma 35. For a counterexample, consider the function $f : \mathbb{N} \to \mathbb{N}$ given by $f(n) = 1$ if $n = k!{+}1$ for some k, and $f(n) = n$ otherwise. Let $m \in \mathbb{N}$. Then $\forall k \geq m, k!{+}1 \equiv 1 (\bmod m)$, and hence $f(n) \equiv n (\bmod m)$ for large enough $n \in \mathbb{N}$. This function f is ultimately periodic modulo every $m \in \mathbb{N}$, but it has no limit. Then $2^{f(n)} \equiv 2 (\bmod 4)$ if and only if n is of the form $n = k! + 1$, thus not periodic.

6 Characterisation of transducts of spiralling sequences

In this section, we characterise the transducts of spiralling sequences in terms of mass products.

The following theorem characterises transducts of spiralling sequences up to equivalence in terms of displaced mass products.

Theorem 37 ([12]) *Let $f \in \circledcirc$, and $\sigma \in \mathbf{2}^{\mathbb{N}}$. We have $\langle f \rangle \geq \sigma$ if and only if*

$$\sigma \equiv \langle \boldsymbol{\beta} \oplus (\overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^{n_0}(f)) \rangle$$

for some $n_0 \in \mathbb{N}$, a mass $\overrightarrow{\boldsymbol{\alpha}}$ and $\boldsymbol{\beta} \in \mathbb{Q}^{|\overrightarrow{\boldsymbol{\alpha}}|}$.

For our purposes, we require a slight strengthening of this theorem ensuring that the transduct is either ultimately periodic or spiralling.

Theorem 38 *Let $f \in \circledcirc$, and $\sigma \in \mathbf{2}^{\mathbb{N}}$. We have $\langle f \rangle \geq \sigma$ if and only if*

(i) σ is ultimately periodic, or
(ii) $\sigma \equiv \langle \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^{n_0}(f) \rangle$ for some $n_0 \in \mathbb{N}$, and a positive mass $\overrightarrow{\boldsymbol{\alpha}}$.

In the latter case, we have $\overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^{n_0}(f) \in \circledcirc$ by Lemma 34.

Proof The direction '$\Leftarrow$' follows from Theorem 37.

For '$\Rightarrow$', let $f \in \circledcirc$ and let $\sigma \in \mathbf{2}^{\mathbb{N}}$ be not ultimately periodic. By Theorem 37 and Lemma 25 we have $\sigma \equiv \langle \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^{n_0}(f) \rangle$ for some $n_0 \in \mathbb{N}$, and a mass $\overrightarrow{\boldsymbol{\alpha}}$. As σ is not ultimately periodic, $\overrightarrow{\boldsymbol{\alpha}}$ contains some non-zero weights. By Theorem 28 we conclude that $\sigma \equiv \langle \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^{n_0}(f) \rangle \equiv \langle \overrightarrow{\boldsymbol{\alpha}'} \otimes \mathcal{S}^{n_0}(f) \rangle$ for some positive mass $\overrightarrow{\boldsymbol{\alpha}'}$. □

Theorem 38 characterises transducts of spiralling sequences up to equivalence. The following theorem strengthens the characterisation (equivalence is replaced by shifts) for the case the transduct is also a spiralling sequence.

Theorem 39 ([19]) *Let $f, g \in \circledcirc$. Then $\langle g \rangle \geq \langle f \rangle$ if and only if some shift of f is a displacement of a mass product of a shift of g, that is:*

$$\mathcal{S}^{n_0}(f) = \boldsymbol{\beta} \oplus (\overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^{m_0}(g)) \tag{2}$$

for some $n_0, m_0 \in \mathbb{N}$, a mass $\overrightarrow{\boldsymbol{\alpha}}$ and $\boldsymbol{\beta} \in \mathbb{Q}^{|\overrightarrow{\boldsymbol{\alpha}}|}$.

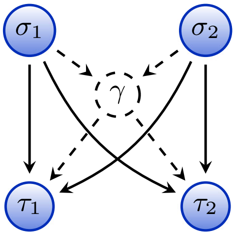


Fig. 7 The idea behind the construction to show that there are pairs of degrees without supremum, and pairs of degrees without infimum; the degree γ does not exist

Theorem 38 states that every transduct of a spiralling stream is either ultimately periodic (in the bottom degree) or again equivalent to a spiralling stream. So, the set of degrees containing spiralling streams is closed under transduction and thus forms a subhierarchy of the Transducer degrees. For understanding the structure of this subhierarchy it suffices to understand when spiralling streams can be transduced into each other. Theorem 39 characterises this inter-transducibility of spiralling streams in terms of displaced mass products:

> For spiralling functions f, g we have $\langle g \rangle \geq \langle f \rangle$ if and only if some shift of f is the displaced mass product of some shift of g.

So, the interesting question is: what preorder does equation (2) induce on the set of spiralling functions?

7 Infima and suprema

After our account of the presence of atom degrees, a next natural focus point of our attention is the question whether the degree hierarchy has suprema and infima for pairs of degrees. Here another contrast with Turing degrees and Mealy degrees manifests itself. While these last hierarchies do possess suprema for every finite set of degrees, we show that this is not the case for Transducer degrees, and thereby answers a question of [17].

We now present two counterexamples for the price of one. We give one construction that yields at the same time that there are pairs of degrees without infimum and pairs of degrees without supremum. The idea of the construction is illustrated in Fig. 7.

We construct streams σ_1, σ_2, τ_1, τ_2 in such a way that the degrees $\sigma_1^{\equiv}$, $\sigma_2^{\equiv}$ are upper bounds of $\{\tau_1^{\equiv}, \tau_2^{\equiv}\}$, and the following property holds:

($\star$) there exists no degree $\gamma^{\equiv}$ that is at the same time a lower bound of $\{\sigma_1^{\equiv}, \sigma_2^{\equiv}\}$ and an upper bound of $\{\tau_1^{\equiv}, \tau_2^{\equiv}\}$.

Observe that any infimum of $\{\sigma_1^{\equiv}, \sigma_2^{\equiv}\}$ or supremum of $\{\tau_1^{\equiv}, \tau_2^{\equiv}\}$ would satisfy this property. Showing that such a the degree $\gamma^{\equiv}$ does not exist, thus implies that $\{\sigma_1^{\equiv}, \sigma_2^{\equiv}\}$ has no infimum, and $\{\tau_1^{\equiv}, \tau_2^{\equiv}\}$ has no supremum.

For convenience, we introduce the following notation:

Definition 40 For sets of streams (or degrees) U, V we write $U \geq V$ if $u \geq v$ for every $u \in U$ and $v \in V$.

We have $\{\sigma_1, \sigma_2\} \geq \{\tau_1, \tau_2\}$, but there does not exist γ such that

$$\{\sigma_1, \sigma_2\} \geq \{\gamma\} \geq \{\tau_1, \tau_2\}. \qquad (\star)$$

To this end, we choose $\sigma_1, \sigma_2, \tau_1, \tau_2$ as follows:

Definition 41 We define $\sigma_1, \sigma_2, \tau_1, \tau_2 \in \{0, 1\}^\omega$ by:

$$\sigma_1 = \prod_{i=0}^{\infty} (10^{2^{2^i}} 10^{3^{3^{3^i}}}) \qquad \sigma_2 = \prod_{i=0}^{\infty} (10^{3^{3^{3^i}}} 10^{2^{2^i}})$$

$$\tau_1 = \prod_{i=0}^{\infty} 10^{2^{2^i}} = \langle 2^{2^i} \rangle \qquad \tau_2 = \prod_{i=0}^{\infty} 10^{3^{3^{3^i}}} = \langle 3^{3^{3^i}} \rangle$$

We have

$$\{\sigma_1, \sigma_2\} \geq \{\tau_1, \tau_2\}$$

by deleting every second block (beginning from the first or second block, respectively). Thus, $\sigma_1^{\equiv}, \sigma_2^{\equiv}$ are upper bounds of $\{\tau_1^{\equiv}, \tau_2^{\equiv}\}$. Figure 8 illustrates the growth of the block sizes in $\sigma_1, \sigma_2, \tau_1, \tau_2$ and indicates possible and impossible transductions between these streams.

Proving ($\star$) is non-trivial as we need to reason about all possible upper/lower bounds, and all possible finite state transductions. To reason about the transducts of σ_1 and σ_2, we will employ Theorems 38 and 39.

Remark 42 Note that the double exponential growth 2^{2^i} of the block sizes is necessary for the construction. Consider

$$\sigma_1' = \prod_{i=0}^{\infty} (10^{2^i} 10^{3^{3^{3^i}}}) \qquad \sigma_2' = \prod_{i=0}^{\infty} (10^{3^{3^{3^i}}} 10^{2^i})$$

$$\tau_1' = \prod_{i=0}^{\infty} 10^{2^i} \qquad \tau_2' = \prod_{i=0}^{\infty} 10^{3^{3^{3^i}}} .$$

Then we have $\sigma_1' \equiv \sigma_2'$, both streams reside in the same degree. For instance, for $\sigma_1' \geq \sigma_2'$: delete the first block 10, and transduce every block 0^{2^i} to $0^{2^{i-1}}$ by replacing $00 \mapsto 0$. Then $\gamma^{\equiv}$ exists, e.g., take $\gamma = \sigma_1'$.

We define functions $s_1, s_2, t_1, t_2 : \mathbb{N} \to \mathbb{N}$ that correspond to the block size of $\sigma_1, \sigma_2, \tau_1, \tau_2$, respectively. We recall that the block size is the stream of run-lengths of blocks of zeros (the distance of ones in the stream).

Definition 43 We define $t_1, t_2 : \mathbb{N} \to \mathbb{N}$ by

$$t_1(n) = 2^{2^n} \qquad t_2(n) = 3^{3^{3^n}},$$

and $s_1, s_2 : \mathbb{N} \to \mathbb{N}$ by

$$s_1(n) = \begin{cases} t_1(n/2) & \text{if } n \text{ is even} \\ t_2((n-1)/2) & \text{if } n \text{ is odd} \end{cases} \qquad s_2(n) = \begin{cases} t_2(n/2) & \text{if } n \text{ is even} \\ t_1((n-1)/2) & \text{if } n \text{ is odd.} \end{cases}$$

Lemma 44 *We have* $\sigma_1 = \langle s_1 \rangle$, $\sigma_2 = \langle s_2 \rangle$, $\tau_1 = \langle t_1 \rangle$ *and* $\tau_2 = \langle t_2 \rangle$. □

Lemma 45 *We have* $s_1, s_2, t_1, t_2 \in$ 🌀. □

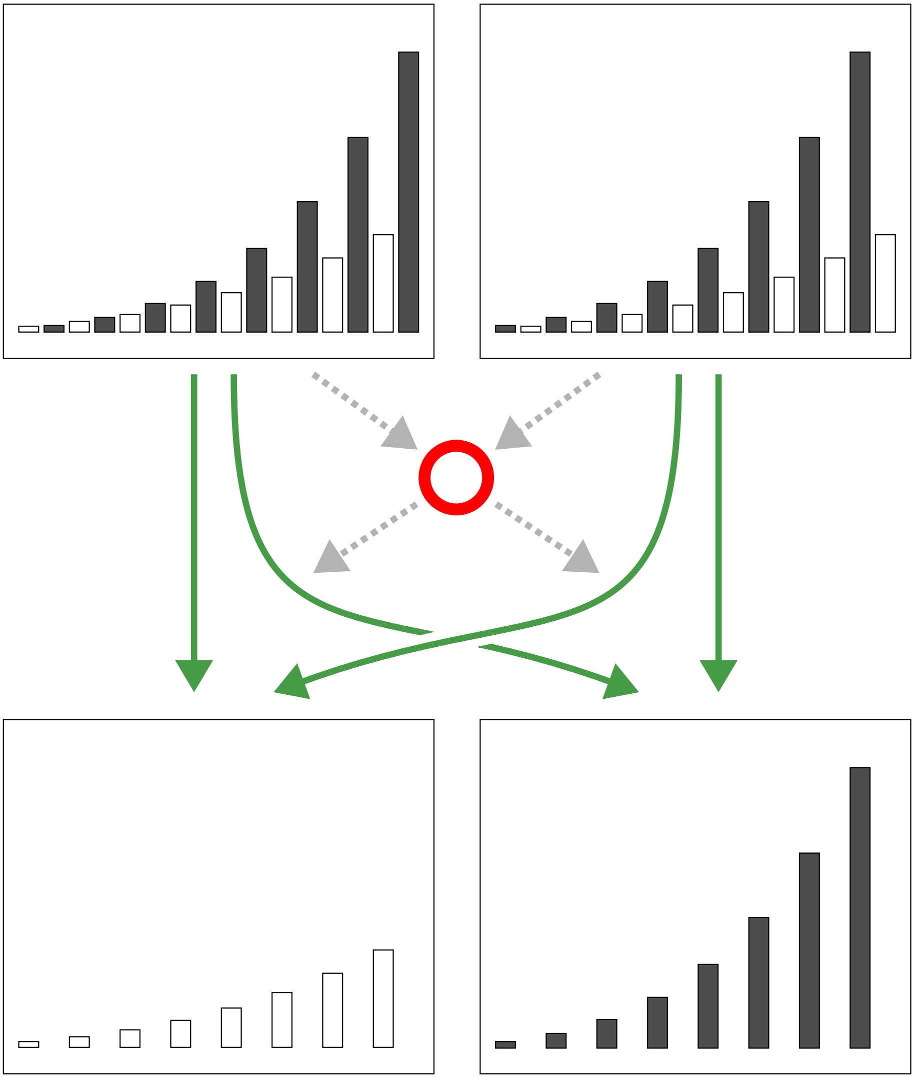


Fig. 8 The graphs illustrate growth of the block size of σ_1 (upper left), σ_2 (upper right), τ_1 (lower left) and τ_2 (lower right). The green (solid) arrows stand for finite state transductions, while the red circle and the grey (dotted) arrows represent impossible transductions (color figure online)

7.1 Proof idea

Before we get to the proof we will give a rough sketch of the idea. Assume that

$$\{\langle s_1\rangle, \langle s_2\rangle\} \geq \{\langle g\rangle\} \geq \{\langle t_1\rangle, \langle t_2\rangle\}$$

for some $g : \mathbb{N} \to \mathbb{N}$. By Theorem 38 we may assume that g is spiralling.

For functions $f, h : \mathbb{N} \to \mathbb{N}$, we write

$$f \approx h \quad \Longleftrightarrow \quad \exists c_1, c_2 > 0.\ \exists n_f, n_h.\ \forall n.\ c_1 h(n_h + n) \le f(n_f + n) \le c_2 h(n_h + n).$$

Note that $\approx$ is an equivalence relation (symmetric and transitive). (The relation $\approx$ is similar to the well-known Big-Theta notation.)

The basic idea of the proof is as follows. We show that

$$\big(\{\langle s_1 \rangle\} \ge \{\langle g \rangle\} \ge \{\langle t_1 \rangle, \langle t_2 \rangle\}\big) \quad \Longrightarrow \quad s_1 \approx g, \tag{3}$$

and similarly,

$$\big(\{\langle s_2 \rangle\} \ge \{\langle g \rangle\} \ge \{\langle t_1 \rangle, \langle t_2 \rangle\}\big) \quad \Longrightarrow \quad s_2 \approx g.$$

Then we have $s_1 \approx g \approx s_2$. However, this yields a contradiction since $s_1 \not\approx s_2$!

We briefly outline the proof of (3). In this sketch we ignore shifts and displacements. By Theorem 39 every spiralling transduct $\langle h \rangle$ of $\langle s_1 \rangle$ is of the form $h = \overrightarrow{\boldsymbol{\alpha}} \otimes s_1$:

$$\begin{tikzcd}
s_1 & 2^{2^0} \arrow[dr, "\alpha|_0"] & 3^{3^{3^0}} \arrow[d, "\alpha|_1"] & 2^{2^1} \arrow[dl, "\alpha|_2"] & 3^{3^{3^1}} \arrow[d, "\alpha|_3"] & 2^{2^2} \arrow[dl, "\alpha|_4"] & 3^{3^{3^2}} \arrow[dr, "\alpha|_0"] & 2^{2^3} \arrow[d, "\alpha|_1"] & 3^{3^{3^3}} \arrow[dl, "\alpha|_2"] & 2^{2^4} \arrow[d, "\alpha|_3"] & 3^{3^{3^4}} \arrow[dl, "\alpha|_4"] & \cdots \\
\overrightarrow{\boldsymbol{\alpha}} \otimes s_1 & & h(0) & & h(1) & & & h(2) & & h(3) & & \cdots
\end{tikzcd}$$

We consider a transduction from $\langle s_1 \rangle$ to $\langle t_1 \rangle$; let $t_1 = \overrightarrow{\boldsymbol{\alpha}} \otimes s_1$. In Fig. 8 this is a transduction from the upper left to the lower left; referring to this figure we will speak of white and black blocks. Then:

(i) The 'only' way to transduce $\langle s_1 \rangle$ to $\langle t_1 \rangle$ is to erase all black blocks of length $3^{3^{3^n}}$, for otherwise the lengths of the blocks of the transduct grows faster than $t_1(n) = 2^{2^n}$. As a consequence $||\overrightarrow{\boldsymbol{\alpha}}||$ is even and $\alpha|_i = 0$ for every odd i.

(ii) Thus, $\langle t_1 \rangle$ is created only from the white blocks of length 2^{2^n} in $\langle s_1 \rangle$. The transduction cannot erase (skip) any of these blocks as the resulting subsequence would grow faster than t_1. Thus, $\alpha|_i > 0$ if and only if i is even.

(iii) For the same reason, the transduction also cannot merge white blocks of length 2^{2^n} in $\langle s_1 \rangle$. So, every weight in $\overrightarrow{\boldsymbol{\alpha}}$ contains precisely one positive $\alpha|_i$ with i even.

For a precise formulation of these properties, see Lemma 48.

Likewise, consider a transduction from $\langle s_1 \rangle$ to $\langle t_2 \rangle$; let $t_2 = \overrightarrow{\boldsymbol{\alpha'}} \otimes s_1$. Using similar reasoning as above, we then can conclude that:

(iv) The transduction cannot erase any of the black blocks of length $3^{3^{3^n}}$. Thus, we have $\alpha'|_i > 0$ for every odd i.

See Lemma 49 for the precise formulation.

Now, if $\langle g \rangle$ exists, then we have $\langle s_1 \rangle \ge \langle g \rangle \ge \{\langle t_1 \rangle, \langle t_2 \rangle\}$. This gives rise to transductions from $\langle s_1 \rangle$ to $\langle t_1 \rangle$ and from $\langle s_1 \rangle$ to $\langle t_2 \rangle$ which both 'start' with common transduction from $\langle s_1 \rangle$ to $\langle g \rangle$. Therefore, consider the common begin piece from $\langle s_1 \rangle$ to $\langle g \rangle$, say $g = \overrightarrow{\boldsymbol{\beta}} \otimes s_1$. Then:

(v) The transduction can neither erase white blocks of length 2^{2^n} by (ii), nor black blocks of length $3^{3^{3^n}}$ by (vi). So, $\beta|_i > 0$ for every i.

(vi) The transduction can also not merge the white blocks of length 2^{2^n} with any other block by (i) and (iii). Thus, every weight in $\overrightarrow{\boldsymbol{\beta}}$ has length 1.

So, the crucial conclusion is: every weight in $\overrightarrow{\boldsymbol{\beta}}$ is non-zero and has length 1! Thus, the transduction $\langle s_1 \rangle \ge \langle g \rangle$ is of the form

$$\begin{tikzcd}[column sep=small]
s_1 & 2^{2^0} \arrow[d, dash, "\beta|_0"] & 3^{3^{3^0}} \arrow[d, dash, "\beta|_1"] & 2^{2^1} \arrow[d, dash, "\beta|_2"] & 3^{3^{3^1}} \arrow[d, dash, "\beta|_3"] & 2^{2^2} \arrow[d, dash, "\beta|_0"] & 3^{3^{3^2}} \arrow[d, dash, "\beta|_1"] & 2^{2^3} \arrow[d, dash, "\beta|_2"] & 3^{3^{3^3}} \arrow[d, dash, "\beta|_3"] & \cdots \\
g & \beta|_0{\cdot}2^{2^0} & \beta|_1{\cdot}3^{3^{3^0}} & \beta|_2{\cdot}2^{2^1} & \beta|_3{\cdot}3^{3^{3^1}} & \beta|_0{\cdot}2^{2^2} & \beta|_1{\cdot}3^{3^{3^2}} & \beta|_2{\cdot}2^{2^3} & \beta|_3{\cdot}3^{3^{3^3}} & \cdots
\end{tikzcd}$$

As a consequence, we have $s_1 \approx g$. Now, the contradiction arises as outlined above.

7.2 Proof

We will now prove the non-existence of γ in several steps. In the next lemmas we investigate how γ arises as a transduct of $\sigma_1 = \langle s_1 \rangle$, and how γ gives rise to $\tau_1 = \langle t_1 \rangle$ and $\tau_2 = \langle t_2 \rangle$.

Lemma 46 *Assume that $\{\sigma_1, \sigma_2\} \geq \{\gamma\} \geq \{\tau_1, \tau_2\}$ for some stream $\gamma \in \mathbf{2}^{\mathbb{N}}$. Then the same holds for a spiralling stream γ of the form*

$$\gamma = \langle \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^{n_0}(s_1) \rangle \tag{4}$$

for some $n_0 \in \mathbb{N}$ and a positive mass $\overrightarrow{\boldsymbol{\alpha}}$.

Proof Let $\gamma \in \mathbf{2}^{\mathbb{N}}$ such that $\{\sigma_1, \sigma_2\} \geq \{\gamma\} \geq \{\tau_1, \tau_2\}$. The stream γ is not ultimately periodic as its transducts τ_1 and τ_2 are not. By Theorem 38 there exists $n_0 \in \mathbb{N}$ and a positive mass $\overrightarrow{\boldsymbol{\alpha}}$ such that $\gamma \equiv \langle \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^{n_0}(s_1) \rangle$. Since equivalent streams behave the same with respect to transducibility, without loss of generality we may assume that $\gamma = \langle \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^{n_0}(s_1) \rangle$. Moreover, by Lemma 34 we have $\overrightarrow{\alpha} \otimes \mathcal{S}^{n_0}(s_1) \in$ 🌀. □

Lemma 47 *Assume that $\{\sigma_1, \sigma_2\} \geq \{\gamma\} \geq \{\tau_1, \tau_2\}$ for some stream $\gamma \in \mathbf{2}^{\mathbb{N}}$. Then we have $\{\sigma_1, \sigma_2\} \geq \{\gamma\} \geq \{\tau_1, \tau_2\}$ for some γ such that*

$$\begin{aligned}
\gamma &= \langle \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^{z}(s_1) \rangle \\
\mathcal{S}^{m}(t_1) &= \boldsymbol{\delta} \oplus ((\overrightarrow{\boldsymbol{\beta}} \otimes \overrightarrow{\boldsymbol{\alpha}}) \otimes \mathcal{S}^{z}(s_1)) \\
\mathcal{S}^{m'}(t_2) &= \boldsymbol{\delta'} \oplus ((\overrightarrow{\boldsymbol{\beta'}} \otimes \overrightarrow{\boldsymbol{\alpha}}^{(r)}) \otimes \mathcal{S}^{z'}(s_1))
\end{aligned}$$

for some $z, z' \in \mathbb{N}$ with $z \leq z'$, positive masses $\overrightarrow{\boldsymbol{\alpha}}$, $\overrightarrow{\boldsymbol{\beta}}$, $\overrightarrow{\boldsymbol{\beta'}}$, displacements $\boldsymbol{\delta} \in \mathbb{Q}^{|\overrightarrow{\boldsymbol{\beta}}|}$ and $\boldsymbol{\delta'} \in \mathbb{Q}^{|\overrightarrow{\boldsymbol{\beta'}}|}$, and $r \in \{0, \ldots, |\overrightarrow{\boldsymbol{\alpha}}| - 1\}$.

Moreover, we may assume that the following conditions hold:

$$\begin{gathered}
|\overrightarrow{\boldsymbol{\alpha}}| \text{ is even, } ||\overrightarrow{\boldsymbol{\beta}}|| = ||\overrightarrow{\boldsymbol{\beta'}}|| = |\overrightarrow{\boldsymbol{\alpha}}|, \\
z' - z \equiv \sum_{i=0}^{r-1} |\boldsymbol{\alpha_i}| (\mathrm{mod}\, ||\overrightarrow{\boldsymbol{\alpha}}||),
\end{gathered}$$

where $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \ldots, \boldsymbol{\alpha_{\ell-1}} \rangle$.

Proof Let $\gamma \in \mathbf{2}^{\mathbb{N}}$ such that $\{\sigma_1, \sigma_2\} \geq \{\gamma\} \geq \{\tau_1, \tau_2\}$. By Lemma 46 we may assume that $\gamma = \langle \overrightarrow{\boldsymbol{\alpha'}} \otimes \mathcal{S}^{n_0}(s_1) \rangle$ for some $n_0 \in \mathbb{N}$ and a positive mass $\overrightarrow{\boldsymbol{\alpha'}}$ such that $\overrightarrow{\alpha'} \otimes \mathcal{S}^{n_0}(s_1) \in$ 🌀.

Since $\gamma \geq \tau_1$ and $\gamma \geq \tau_2$, Theorem 39 yields

$$\begin{aligned}
\mathcal{S}^{m}(t_1) &= \boldsymbol{\delta} \oplus (\overrightarrow{\boldsymbol{\beta}} \otimes \mathcal{S}^{n_1}(\overrightarrow{\boldsymbol{\alpha'}} \otimes \mathcal{S}^{n_0}(s_1))), \\
\mathcal{S}^{m'}(t_2) &= \boldsymbol{\delta'} \oplus (\overrightarrow{\boldsymbol{\beta'}} \otimes \mathcal{S}^{n_2}(\overrightarrow{\boldsymbol{\alpha'}} \otimes \mathcal{S}^{n_0}(s_1))).
\end{aligned} \tag{5}$$

for some $m, m', n_1, n_2 \in \mathbb{N}$, masses $\overrightarrow{\boldsymbol{\beta}}$, $\overrightarrow{\boldsymbol{\beta}'}$ and $\boldsymbol{\delta} \in \mathbb{Q}^{|\overrightarrow{\boldsymbol{\beta}}|}$, $\boldsymbol{\delta}' \in \mathbb{Q}^{|\overrightarrow{\boldsymbol{\beta}'}|}$. Without loss of generality $n_1 \leq n_2$, for otherwise we can take shifts (of left- and right-hand side) in (5). Moreover, we may assume that

(i) $|\overrightarrow{\boldsymbol{\alpha}'}|$ is even, for otherwise we can use $\overrightarrow{\boldsymbol{\alpha}'}^{2}$ instead of $\overrightarrow{\boldsymbol{\alpha}'}$, and
(ii) $||\overrightarrow{\boldsymbol{\beta}}|| = ||\overrightarrow{\boldsymbol{\beta}'}|| = |\overrightarrow{\boldsymbol{\alpha}'}|$, for otherwise we can unfold $\overrightarrow{\boldsymbol{\alpha}'}$, $\overrightarrow{\boldsymbol{\beta}}$, $\overrightarrow{\boldsymbol{\beta}'}$, $\boldsymbol{\delta}$ and $\boldsymbol{\delta}'$ to the least common multiple of $||\boldsymbol{\beta}||$, $||\overrightarrow{\boldsymbol{\beta}'}||$, $|\overrightarrow{\boldsymbol{\alpha}'}|$.

There exist unique $h_1, h_2 \in \mathbb{N}$ and $h'_1, h'_2 \in \{0, \ldots, |\overrightarrow{\boldsymbol{\alpha}'}| - 1\}$ such that

$$n_1 = h_1|\overrightarrow{\boldsymbol{\alpha}'}| + h'_1 \qquad\qquad n_2 = h_2|\overrightarrow{\boldsymbol{\alpha}'}| + h'_2.$$

Let $\overrightarrow{\boldsymbol{\alpha}'} = \langle \boldsymbol{\alpha}'_{\mathbf{0}}, \ldots, \boldsymbol{\alpha}'_{\boldsymbol{\ell-1}} \rangle$ and define

$$z = n_0 + h_1||\overrightarrow{\boldsymbol{\alpha}'}|| + \sum_{i=0}^{h'_1-1} |\boldsymbol{\alpha}'_i|, \qquad\qquad z' = n_0 + h_2||\overrightarrow{\boldsymbol{\alpha}'}|| + \sum_{i=0}^{h'_2-1} |\boldsymbol{\alpha}'_i|.$$

From $n_1 \leq n_2$ follows that $h_1 \leq h_2$ and $z \leq z'$. Then by Lemma 19, we have

$$\mathcal{S}^m(t_1) = \boldsymbol{\delta} \oplus (\overrightarrow{\boldsymbol{\beta}} \otimes (\overrightarrow{\boldsymbol{\alpha}'}^{(h'_1)} \otimes \mathcal{S}^z(s_1))),$$
$$\mathcal{S}^{m'}(t_2) = \boldsymbol{\delta}' \oplus (\overrightarrow{\boldsymbol{\beta}'} \otimes (\overrightarrow{\boldsymbol{\alpha}'}^{(h'_2)} \otimes \mathcal{S}^{z'}(s_1))).$$

Let $\overrightarrow{\boldsymbol{\alpha}} = \overrightarrow{\boldsymbol{\alpha}'}^{(h'_1)}$. Note that $|\overrightarrow{\boldsymbol{\alpha}}| = |\overrightarrow{\boldsymbol{\alpha}'}|$, and thus, properties (i) and (ii) are not affected. Let $r \in \{0, \ldots, |\overrightarrow{\boldsymbol{\alpha}}| - 1\}$ such that $h'_1 + r \equiv h'_2 (\mathrm{mod}\, |\overrightarrow{\boldsymbol{\alpha}}|)$. Then $\overrightarrow{\boldsymbol{\alpha}}^{(r)} = \overrightarrow{\boldsymbol{\alpha}'}^{(h'_2)}$ and we have

$$\mathcal{S}^m(t_1) = \boldsymbol{\delta} \oplus ((\overrightarrow{\boldsymbol{\beta}} \otimes \overrightarrow{\boldsymbol{\alpha}}) \otimes \mathcal{S}^z(s_1)),$$
$$\mathcal{S}^{m'}(t_2) = \boldsymbol{\delta}' \oplus ((\overrightarrow{\boldsymbol{\beta}'} \otimes \overrightarrow{\boldsymbol{\alpha}}^{(r)}) \otimes \mathcal{S}^{z'}(s_1)) \tag{6}$$

by Lemma 24. Moreover, $\langle \gamma \rangle \equiv \langle \mathcal{S}^{n_1}(\gamma) \rangle$ and by Lemma 19:

$$\mathcal{S}^{n_1}(\gamma) = \mathcal{S}^{n_1}(\overrightarrow{\boldsymbol{\alpha}'} \otimes \mathcal{S}^{n_0}(s_1)) = \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^z(s_1).$$

So, without loss of generality, we can take $\gamma = \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^z(s_1)$.

Since $h_1 \leq h_2$,

$$z - z' \equiv \sum_{i=0}^{h'_1-1} |\boldsymbol{\alpha}'_i| - \sum_{i=0}^{h'_2-1} |\boldsymbol{\alpha}'_i| \equiv -\sum_{i=h'_1}^{h'_2-1} |\boldsymbol{\alpha}'_i| \equiv -\sum_{i=0}^{r-1} |\boldsymbol{\alpha}_i| (\mathrm{mod}\, ||\overrightarrow{\boldsymbol{\alpha}}||).$$

This concludes the proof. □

The function s_1 is an alternating interleaving of $t_1(n) = 2^{2^n}$ and $t_2(n) = 3^{3^{3^n}}$; we will speak of white and black blocks, respectively. So, the white blocks are those at even positions, and the black blocks are at the odd positions in $\langle s_1 \rangle$. The following lemma investigates how $\langle t_1 \rangle$ can arise as a transduct of $\langle s_1 \rangle$. It states that, any transduction that transforms $\langle s_1 \rangle$ into $\langle t_1 \rangle$ has the following properties:

(a) It must erase all black blocks; so, all the blocks of length $3^{3^{3^n}}$ must be multiplied by 0.
(b) It cannot erase (or merge) any of the white blocks of length 2^{2^n}.

A violation of (a) or (b) would result in a transduct whose block size grows too quickly, that is, more than a constant factor faster than t_1.

Lemma 48 *Assume that*

$$\mathcal{S}^m(t_1) = \boldsymbol{\delta} \oplus (\overrightarrow{\boldsymbol{\tau}} \otimes \mathcal{S}^z(s_1)) \tag{7}$$

for some $m, z \in \mathbb{N}$, $\overrightarrow{\boldsymbol{\tau}} = \langle \boldsymbol{\tau_0}, \dots, \boldsymbol{\tau_{\ell'-1}} \rangle$ *a mass and* $\boldsymbol{\delta} \in \mathbb{Q}^{|\overrightarrow{\tau}|}$. *Moreover, assume that* $||\overrightarrow{\boldsymbol{\tau}}||$ *is even. Then:*

(i) $\forall i \in \{0, \dots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\}$: $\tau|_i = 0 \iff z + i$ *is odd.*
(ii) If z *is even, then* $\forall i \in \{0, \dots, \ell' - 1\}$: $\varphi(\overrightarrow{\boldsymbol{\tau}}, 2i) = i$.
(iii) If z *is odd, then* $\forall i \in \{0, \dots, \ell' - 1\}$: $\varphi(\overrightarrow{\boldsymbol{\tau}}, 2i + 1) = i$.

Proof We start by showing that:

$$\tau|_i = 0 \text{for every } i \in \{0, \dots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\} \text{such that } z + i \text{ is odd.} \tag{8}$$

For a contradiction, assume that $\tau|_k \neq 0$ for some $k \in \{0, \dots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\}$ such that $z + k$ is odd. Let $\boldsymbol{\delta} = \langle \delta_0, \dots, \delta_{\ell'-1} \rangle$.

From assumption (7) and Lemma 16 we obtain:

$$\mathcal{S}^m(t_1)(h|\overrightarrow{\boldsymbol{\tau}}| + h') = \delta_{h'} + \sum_{\substack{i \in \{0, \dots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\} \\ \varphi(\overrightarrow{\boldsymbol{\tau}}, i) = h'}} \big(\tau|_i \cdot \mathcal{S}^z(s_1)(h||\overrightarrow{\boldsymbol{\tau}}|| + i) \big)$$

for every $h \in \mathbb{N}$ and $h' \in \{0, \dots, |\overrightarrow{\boldsymbol{\tau}}| - 1\}$. Take $h' = \varphi(\overrightarrow{\boldsymbol{\tau}}, k)$, then

$$\mathcal{S}^m(t_1)(h|\overrightarrow{\boldsymbol{\tau}}| + h') \geq \tau|_k \cdot \mathcal{S}^z(s_1)(h||\overrightarrow{\boldsymbol{\tau}}|| + k)$$

and, hence,

$$t_1(m + h|\overrightarrow{\boldsymbol{\tau}}| + h') \geq \tau|_k \cdot s_1(z + h||\overrightarrow{\boldsymbol{\tau}}|| + k) = \tau|_k \cdot t_2(\frac{z + h||\overrightarrow{\boldsymbol{\tau}}|| + k - 1}{2}) \tag{9}$$

since $z + k$ is odd and $||\overrightarrow{\boldsymbol{\tau}}||$ is even. Recall that $t_1(n) = 2^{2^n}$ and $t_2(n) = 3^{3^{3^n}}$. So, for large enough h, the formula on the right grows much faster than that on the left-hand side of the inequality. Thus, the inequality (9) cannot hold for all $h \in \mathbb{N}$. This is a contradiction, and we have established that (8) holds.

Thus, every second entry of $\boldsymbol{\tau_0}$; $\cdots$; $\boldsymbol{\tau_{\ell'-1}}$ is 0. However, none of the weights $\boldsymbol{\tau_0}$, ..., $\boldsymbol{\tau_{\ell'-1}}$ is zero since the left-hand side of (7) grows to infinity. So, each of these weights must contain a non-zero entry. It follows that $|\overrightarrow{\boldsymbol{\tau}}| \leq ||\overrightarrow{\boldsymbol{\tau}}||/2$.

Next, we show that:

$$|\overrightarrow{\boldsymbol{\tau}}| = ||\overrightarrow{\boldsymbol{\tau}}||/2. \tag{10}$$

For a contradiction, assume $|\overrightarrow{\boldsymbol{\tau}}| < ||\overrightarrow{\boldsymbol{\tau}}||/2$. Let $\ell_0 = |\boldsymbol{\tau_0}|$. So,

$$\boldsymbol{\tau_0} = \langle \tau|_0, \tau|_1, \dots, \tau|_{\ell_0 - 1} \rangle$$

Since $\boldsymbol{\tau_0}$ is not zero, it holds that $\tau|_k > 0$ for some $k \in \{0, \dots, \ell_0 - 1\}$. Then $z + k$ is even by (8). Let $h' = \varphi(\overrightarrow{\boldsymbol{\tau}}, k)$, then as above we have for every $h \in \mathbb{N}$:

$$t_1(m + h|\overrightarrow{\boldsymbol{\tau}}| + h') \geq \tau|_k \cdot s_1(z + h||\overrightarrow{\boldsymbol{\tau}}|| + k) = \tau|_k \cdot t_1(\frac{z + h||\overrightarrow{\boldsymbol{\tau}}|| + k}{2})$$

since $z + k$ is even and $||\vec{\boldsymbol{\tau}}||$ is even. Therefore,

$$2^{2^{m+h|\vec{\boldsymbol{\tau}}|+h'}} \geq \tau|_k \cdot 2^{2^{\frac{z}{2}+h\frac{||\vec{\boldsymbol{\tau}}||}{2}+\frac{k}{2}}}.$$

However, $|\vec{\boldsymbol{\tau}}| < ||\vec{\boldsymbol{\tau}}||/2$. Hence, the inequality cannot hold for arbitrary large h. This is a contradiction, and, thus, the claim (10) holds.

Let us summarise. Every second entry of $\boldsymbol{\tau_0}\ ;\ \cdots\ ;\ \boldsymbol{\tau_{\ell'-1}}$ is 0 by (8), and we have $\ell' = |\vec{\boldsymbol{\tau}}| = ||\vec{\boldsymbol{\tau}}||/2$ by (10). Thus, $\boldsymbol{\tau_0}\ ;\ \cdots\ ;\ \boldsymbol{\tau_{\ell'-1}}$ contains at most ℓ' non-zero entries. On the other hand, each weight $\boldsymbol{\tau_0}, \ldots, \boldsymbol{\tau_{\ell'-1}}$ contains a non-zero entry. Therefore, we conclude that $\boldsymbol{\tau_0}\ ;\ \cdots\ ;\ \boldsymbol{\tau_{\ell'-1}}$ contains precisely ℓ' non-zero entries, and each weight $\boldsymbol{\tau_0}, \ldots, \boldsymbol{\tau_{\ell'-1}}$ contains precisely one them. From (8) we obtain:

$$\tau|_i = 0 \quad \Longleftrightarrow \quad z + i \text{ is odd}$$

for every $i \in \{0, \ldots, ||\vec{\boldsymbol{\tau}}|| - 1\}$. Since each weight contains precisely one non-zero entry, we conclude that:

(i) If z is even, then $\forall i \in \{0, \ldots, \ell' - 1\}$: $\boldsymbol{\tau_i}$ contains $\tau|_{2i}$.
(ii) If z is odd, then $\forall i \in \{0, \ldots, \ell' - 1\}$: $\boldsymbol{\tau_i}$ contains $\tau|_{2i+1}$.

This concludes the proof. □

The following lemma investigates how t_2 can be obtained as a transduct of s_1. Roughly speaking, the lemma states that the values of s_1 at odd indices are copied, giving rise to t_2.

Lemma 49 *Assume that*

$$\mathcal{S}^m(t_2) = \boldsymbol{\delta} \oplus (\vec{\boldsymbol{\tau}} \otimes \mathcal{S}^z(s_1)) \tag{11}$$

for some $m, z \in \mathbb{N}$, $\vec{\boldsymbol{\tau}} = \langle \boldsymbol{\tau_0}, \ldots, \boldsymbol{\tau_{\ell'-1}} \rangle$ a mass and $\boldsymbol{\delta} \in \mathbb{Q}^{|\tau|}$. Moreover, assume that $||\vec{\boldsymbol{\tau}}||$ is even. Then:

(i) $\forall i \in \{0, \ldots, ||\vec{\boldsymbol{\tau}}|| - 1\}$: $\tau|_i > 0$ if $z + i$ is odd.
(ii) If z is even, then $\forall i \in \{0, \ldots, \ell' - 1\}$: $\varphi(\vec{\boldsymbol{\tau}}, 2i + 1) = i$.
(iii) If z is odd, then$\forall i \in \{0, \ldots, \ell' - 1\}$: $\varphi(\vec{\boldsymbol{\tau}}, 2i) = i$.

Proof First, we show that every weight $\boldsymbol{\tau_0}, \ldots, \boldsymbol{\tau_{\ell-1}}$ contains some $\tau|_i$ such that $\tau|_i > 0$ and $z + i$ is odd. More precisely, we prove:

$$\begin{aligned} &\forall h' \in \{0, \ldots, |\vec{\boldsymbol{\tau}}| - 1\}.\ \exists i \in \{0, \ldots, ||\vec{\boldsymbol{\tau}}|| - 1\}. \\ &\qquad \varphi(\vec{\boldsymbol{\tau}}, i) = h' \ \wedge\ \tau|_i > 0 \ \wedge\ z + i \text{ is odd} \end{aligned} \tag{12}$$

For a contradiction, we assume that there exists $h' \in \{0, \ldots, |\vec{\boldsymbol{\tau}}| - 1\}$ such that

(†) $\tau|_i = 0$ for every $i \in \{0, \ldots, ||\vec{\boldsymbol{\tau}}|| - 1\}$ for which $\varphi(\vec{\boldsymbol{\tau}}, i) = h'$ and $z + i$ is odd.

From assumption (11) and Lemma 16 we obtain:

$$\begin{aligned} \mathcal{S}^m(t_2)(h|\vec{\boldsymbol{\tau}}| + h') &= \delta_{h'} + \sum_{\substack{i \in \{0, \ldots, ||\vec{\boldsymbol{\tau}}||-1\} \\ \varphi(\vec{\boldsymbol{\tau}}, i) = h'}} \big(\tau|_i \cdot s_1(z + h||\vec{\boldsymbol{\tau}}|| + i) \big) \\ &= \delta_{h'} + \sum_{\substack{i \in \{0, \ldots, ||\vec{\boldsymbol{\tau}}||-1\} \\ \varphi(\vec{\boldsymbol{\tau}}, i) = h' \\ z+i \text{ is even}}} \big(\tau|_i \cdot t_1(\frac{z + h||\vec{\boldsymbol{\tau}}|| + i}{2}) \big) \quad (\text{by}(\dagger)) \end{aligned}$$

for every $h \in \mathbb{N}$ since $||\overrightarrow{\boldsymbol{\tau}}||$ is even. Thus, for

$$c = \sum_{\substack{i \in \{0,\ldots,||\overrightarrow{\boldsymbol{\tau}}||-1\} \\ \varphi(\overrightarrow{\boldsymbol{\tau}},i)=h' \\ z+i \text{ is even}}} \tau|_i ,$$

we have

$$t_2(m + h|\overrightarrow{\boldsymbol{\tau}}| + h') \leq \delta_{h'} + c \cdot t_1(\frac{z + h||\overrightarrow{\boldsymbol{\tau}}|| + ||\overrightarrow{\boldsymbol{\tau}}||}{2}) \tag{13}$$

for every $h \in \mathbb{N}$. Recall that $t_1(n) = 2^{2^n}$ and $t_2(n) = 3^{3^{3^n}}$. Thus, for large enough h, the inequality (13) cannot hold. This is a contradiction, and hence, claim (12) holds.

As $||\overrightarrow{\boldsymbol{\tau}}||$ is even, from (12) it follows that $|\overrightarrow{\boldsymbol{\tau}}| \leq ||\overrightarrow{\boldsymbol{\tau}}||/2$. We show that:

$$|\overrightarrow{\boldsymbol{\tau}}| = ||\overrightarrow{\boldsymbol{\tau}}||/2. \tag{14}$$

For a contradiction, assume $|\overrightarrow{\boldsymbol{\tau}}| < ||\overrightarrow{\boldsymbol{\tau}}||/2$. Let $\ell_0 = |\boldsymbol{\tau_0}|$. So,

$$\boldsymbol{\tau_0} = \langle \tau|_0, \tau|_1, \ldots, \tau|_{\ell_0 - 1} \rangle.$$

By (12), $\tau|_k > 0$ for some $k \in \{0, \ldots, \ell_0 - 1\}$ such that $z + k$ is odd. Let $h' = \varphi(\overrightarrow{\boldsymbol{\tau}}, k)$. From assumption (11) and Lemma 16 it follows that:

$$t_2(m + h|\overrightarrow{\boldsymbol{\tau}}| + h') \geq \tau|_k \cdot s_1(z + h||\overrightarrow{\boldsymbol{\tau}}|| + k) = \tau|_k \cdot t_2(\frac{z + h||\overrightarrow{\boldsymbol{\tau}}|| + k - 1}{2})$$

for every $h \in \mathbb{N}$, and hence,

$$t_2(m + h' + h|\overrightarrow{\boldsymbol{\tau}}|) \geq \tau|_k \cdot t_2(\frac{z + k - 1}{2} + h\frac{||\overrightarrow{\boldsymbol{\tau}}||}{2}).$$

However, $t_2(n) = 3^{3^{3^n}}$ and we have assumed $|\overrightarrow{\boldsymbol{\tau}}| < ||\overrightarrow{\boldsymbol{\tau}}||/2$. Thus, the inequality cannot hold for all $h \in \mathbb{N}$. This is a contradiction, and hence the claim (14) holds.

So, $||\overrightarrow{\boldsymbol{\tau}}||$ is even, $\ell' = |\overrightarrow{\boldsymbol{\tau}}| = ||\overrightarrow{\boldsymbol{\tau}}||/2$ and every weight $\boldsymbol{\tau_0}$, ..., $\boldsymbol{\tau_{\ell'-1}}$ contains some $\tau|_i$ such that $z + i$ is odd. There are precisely $||\overrightarrow{\boldsymbol{\tau}}||/2$ elements $\tau|_i$ such that $z + i$ is odd, and thus each weight contains precisely one of them. This implies that

(i) If z is even, then $\forall i \in \{0, \ldots, \ell' - 1\}$: $\boldsymbol{\tau_i}$ contains $\tau|_{2i}$.
(ii) If z is odd, then $\forall i \in \{0, \ldots, \ell' - 1\}$: $\boldsymbol{\tau_i}$ contains $\tau|_{2i+1}$.

Moreover, using (12) we can conclude that

$$\tau|_i > 0 \quad \iff \quad z + i \text{ is odd} \tag{15}$$

for every $i \in \{0, \ldots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\}$. □

Lemma 50 *There exists no $\gamma \in \mathbf{2}^{\mathbb{N}}$ such that $\{\sigma_1, \sigma_2\} \geq \{\gamma\} \geq \{\tau_1, \tau_2\}$.*

Proof For a contradiction, assume that such $\gamma \in \mathbf{2}^{\mathbb{N}}$ exists. Then, by Lemma 47, we have $\{\sigma_1, \sigma_2\} \geq \{\gamma\} \geq \{\tau_1, \tau_2\}$ for some γ such that

$$\gamma = \langle \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^z(s_1) \rangle$$
$$\mathcal{S}^m(t_1) = \boldsymbol{\delta} \oplus ((\overrightarrow{\boldsymbol{\beta}} \otimes \overrightarrow{\boldsymbol{\alpha}}) \otimes \mathcal{S}^z(s_1))$$
$$\mathcal{S}^{m'}(t_2) = \boldsymbol{\delta'} \oplus ((\overrightarrow{\boldsymbol{\beta'}} \otimes \overrightarrow{\boldsymbol{\alpha}}^{(r)}) \otimes \mathcal{S}^{z'}(s_1))$$

for some $z, z' \in \mathbb{N}$ with $z \le z'$, positive masses $\overrightarrow{\boldsymbol{\alpha}}$, $\overrightarrow{\boldsymbol{\beta}}$, $\overrightarrow{\boldsymbol{\beta'}}$, $\boldsymbol{\delta} \in \mathbb{Q}^{|\overrightarrow{\boldsymbol{\beta}}|}$ and $\boldsymbol{\delta'} \in \mathbb{Q}^{|\overrightarrow{\boldsymbol{\beta'}}|}$, and $r \in \{0, \dots, |\overrightarrow{\boldsymbol{\alpha}}| - 1\}$.

Moreover, we may assume that $|\overrightarrow{\boldsymbol{\alpha}}|$ is even, $||\overrightarrow{\boldsymbol{\beta}}|| = ||\overrightarrow{\boldsymbol{\beta'}}|| = |\overrightarrow{\boldsymbol{\alpha}}|$ and

$$z' - z \equiv R (\mathrm{mod}\, ||\overrightarrow{\boldsymbol{\alpha}}||),$$

where $\overrightarrow{\boldsymbol{\alpha}} = \langle \boldsymbol{\alpha_0}, \dots, \boldsymbol{\alpha_{\ell-1}} \rangle$ and $R = \sum_{i=0}^{r-1} |\boldsymbol{\alpha_i}|$.

The proof now proceeds in three parts, namely,

(A) We show that $\alpha|_i > 0$ for every $i \in \{0, \dots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\}$.
(B) We show that $|\boldsymbol{\alpha_i}| = 1$ for every $i \in \{0, \dots, \ell - 1\}$.
(C) We put things together and conclude.

Part (A): Let $\overrightarrow{\boldsymbol{\tau}} = \overrightarrow{\boldsymbol{\beta}} \otimes \overrightarrow{\boldsymbol{\alpha}}$ and $\overrightarrow{\boldsymbol{\tau'}} = \overrightarrow{\boldsymbol{\beta'}} \otimes \overrightarrow{\boldsymbol{\alpha}}^{(r)}$. Then

$$\mathcal{S}^m(t_1) = \boldsymbol{\delta} \oplus (\overrightarrow{\boldsymbol{\tau}} \otimes \mathcal{S}^z(s_1))$$
$$\mathcal{S}^{m'}(t_2) = \boldsymbol{\delta'} \oplus (\overrightarrow{\boldsymbol{\tau'}} \otimes \mathcal{S}^{z'}(s_1)).$$

By definition of $\otimes$, $||\overrightarrow{\boldsymbol{\tau}}|| = ||\overrightarrow{\boldsymbol{\tau'}}|| = ||\overrightarrow{\boldsymbol{\alpha}}||$ and, for every $i \in \{0, \dots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\}$,

$$\tau|_i = \beta|_{\varphi(\overrightarrow{\boldsymbol{\alpha}}, i)} \cdot \alpha|_i \tag{16}$$
$$\tau'|_i = \beta'|_{\varphi(\overrightarrow{\boldsymbol{\alpha}}^{(r)}, i)} \cdot \alpha^{(r)}|_i. \tag{17}$$

By Lemma 48,

$$\tau|_i > 0 \iff z + i \text{ is even} \tag{18}$$

for every $i \in \{0, \dots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\}$; thus, by (16),

$$z + i \text{ is even} \implies \alpha|_i > 0. \tag{19}$$

By Lemma 49, $z' + j$ is odd $\implies \tau'|_j > 0$ for every $j \in \{0, \dots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\}$; thus, by (17),

$$z' + j \text{ is odd} \implies \alpha^{(r)}|_j > 0. \tag{20}$$

By the definitions of R and $\alpha^{(r)}|_j$,

$$\begin{aligned} \alpha|_i = \alpha^{(r)}|_j &\Leftarrow i \equiv j + R (\mathrm{mod}\, ||\overrightarrow{\boldsymbol{\alpha}}||) \\ &\Leftrightarrow i \equiv j + z' - z (\mathrm{mod}\, ||\overrightarrow{\boldsymbol{\alpha}}||) \\ &\Leftrightarrow z + i \equiv z' + j (\mathrm{mod}\, ||\overrightarrow{\boldsymbol{\alpha}}||) \end{aligned} \tag{21}$$

By (20), (21) and since $||\overrightarrow{\boldsymbol{\alpha}}||$ is even,

$$z + i \text{ is odd} \implies \alpha|_i > 0. \tag{22}$$

for every $i \in \{0, \dots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\}$.

From (19) and (22) it follows that:

$$\forall i \in \{0, \dots, ||\overrightarrow{\boldsymbol{\tau}}|| - 1\}.\ \alpha|_i > 0. \tag{23}$$

Part (B): We show that

$$\forall i \in \{0, \dots, \ell - 1\}.\ |\boldsymbol{\alpha_i}| = 1. \tag{24}$$

For a contradiction, assume that $|\boldsymbol{\alpha_k}| > 1$ for some $k \in \{0, \dots, \ell - 1\}$. Let

$$\begin{aligned} k_1 &= |\boldsymbol{\alpha_0}| + |\boldsymbol{\alpha_1}| + \cdots + |\boldsymbol{\alpha_{k-1}}|, \\ k_2 &= k_1 + |\boldsymbol{\alpha_k}| - 1. \end{aligned}$$

Then $k_1 < k_2$, $\boldsymbol{\alpha_k} = \langle \alpha|_{k_1}, \dots, \alpha|_{k_2} \rangle$, and $\varphi(\overrightarrow{\boldsymbol{\alpha}}, i) = k$ for every $i \in \{k_1, \dots, k_2\}$. Let $h, h' \in \{k_1, \dots, k_2\}$ such that $z + h$ is even and $z + h'$ is odd. Since $z + h$ is even, we get $\tau|_h > 0$ from (18). As $\varphi(\overrightarrow{\boldsymbol{\alpha}}, h) = k$, we conclude $\beta|_k > 0$ by (16). However, by (16) and (23), $\tau|_i > 0$ for every $i \in \{k_1, \dots, k_2\}$; thus also for $i = h'$. This contradicts (18) since $z + h'$ is odd. Hence, claim (24) holds.

Part (C): Recall that

$$\gamma = \langle \overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^z(s_1) \rangle.$$

By Theorem 39,

$$\mathcal{S}^{n_0}(\overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^z(s_1)) = \boldsymbol{\delta} \oplus (\overrightarrow{\boldsymbol{\xi}} \otimes \mathcal{S}^{m_0}(s_2))$$

for some $n_0, m_0 \in \mathbb{N}$, a mass $\overrightarrow{\boldsymbol{\xi}} = \langle \boldsymbol{\xi_0}, \dots, \boldsymbol{\xi_{\ell'-1}} \rangle$ and $\boldsymbol{\delta} \in \mathbb{Q}^{|\overrightarrow{\boldsymbol{\xi}}|}$.

In parts (A) and (B) of this proof, we have established that $|\boldsymbol{\alpha_i}| = 1$ for every $i \in \{0, \dots, \ell - 1\}$. As the functions s_1 and s_2 are both obtained by interleaving t_1 and t_2, we can apply the same reasoning to transductions of $\langle s_2 \rangle$ as we have done for $\langle s_1 \rangle$. So, by reasoning as above (using analogous versions of Lemmas 48 and 49), we conclude that $|\boldsymbol{\xi_i}| = 1$ for every $i \in \{0, \dots, \ell' - 1\}$.

By Lemma 19, we get

$$\mathcal{S}^{n_0}(\overrightarrow{\boldsymbol{\alpha}} \otimes \mathcal{S}^z(s_1)) = \overrightarrow{\boldsymbol{\alpha}}^{(n_0)} \otimes \mathcal{S}^{z+n_0}(s_1) = \boldsymbol{\delta} \oplus (\overrightarrow{\boldsymbol{\xi}} \otimes \mathcal{S}^{m_0}(s_2))$$

since $|\boldsymbol{\alpha_i}| = 1$ for every $i \in \{0, \dots, \ell - 1\}$.

Thus,

$$\overrightarrow{\boldsymbol{\zeta}} \otimes \mathcal{S}^n(s_1) = \boldsymbol{\delta} \oplus (\overrightarrow{\boldsymbol{\xi}} \otimes \mathcal{S}^m(s_2)) \tag{25}$$

for $\zeta = \overrightarrow{\boldsymbol{\alpha}}^{(n_0)}$, $n = z + n_0$ and $m = m_0$. Both $\overrightarrow{\boldsymbol{\zeta}}$, $\overrightarrow{\boldsymbol{\xi}}$ are positive masses consisting only of weights of length 1, and $|\boldsymbol{\delta}| = |\overrightarrow{\boldsymbol{\xi}}|$.

Without loss of generality we may assume that:

(i) n is even, for otherwise we can take a shift $\mathcal{S}(\,\cdot\,)$ on left and right;
(ii) $|\overrightarrow{\boldsymbol{\zeta}}| = |\overrightarrow{\boldsymbol{\xi}}| = |\boldsymbol{\delta}|$ and $|\overrightarrow{\boldsymbol{\zeta}}|$ is even; otherwise, we unfold $\overrightarrow{\boldsymbol{\zeta}}$, $\overrightarrow{\boldsymbol{\xi}}$ and $\boldsymbol{\delta}$.

Then

$$\begin{aligned} \overrightarrow{\boldsymbol{\zeta}} &= \langle \langle \zeta_0 \rangle, \dots, \langle \zeta_{\ell-1} \rangle \rangle \\ \overrightarrow{\boldsymbol{\xi}} &= \langle \langle \xi_0 \rangle, \dots, \langle \xi_{\ell-1} \rangle \rangle \\ \boldsymbol{\delta} &= \langle \langle \delta_0 \rangle, \dots, \langle \delta_{\ell-1} \rangle \rangle \end{aligned}$$

for some even $\ell \in \mathbb{N}$, $\zeta_0, \dots, \zeta_{\ell-1} > 0$ and $\xi_0, \dots, \xi_{\ell-1} > 0$. From (25), we get

$$(\overrightarrow{\boldsymbol{\zeta}} \otimes \mathcal{S}^n(s_1))(h\ell) = (\boldsymbol{\delta} \oplus (\overrightarrow{\boldsymbol{\xi}} \otimes \mathcal{S}^m(s_2)))(h\ell)$$

for every $h \in \mathbb{N}$. We have

$$(\overrightarrow{\boldsymbol{\zeta}} \otimes \mathcal{S}^n(s_1))(h\ell) = \zeta_0 \cdot t_1((n + h\ell)/2)$$
$$(\boldsymbol{\delta} \oplus (\overrightarrow{\boldsymbol{\xi}} \otimes \mathcal{S}^m(s_2)))(h\ell) = \delta_0 + \xi_0 \cdot s_2(m + h\ell)$$

and, hence,

$$\zeta_0 \cdot t_1((n + h\ell)/2) = \delta_0 + \xi_0 \cdot s_2(m + h\ell)$$

for every $h \in \mathbb{N}$. If m is even, then $s_2(m + h\ell) = t_2((m + h\ell)/2)$. This gives a contradiction since t_2 grows much faster than t_1. Thus m must be odd, and

$$\begin{aligned} &\zeta_0 \cdot t_1((n + h\ell)/2) = \zeta_0 \cdot 2^{2^{(n+h\ell)/2}} \\ &\qquad\qquad \| \\ &\delta_0 + \xi_0 \cdot s_2(m + h\ell) = \delta_0 + \xi_0 \cdot t_1((m + h\ell - 1)/2) = \delta_0 + \xi_0 \cdot 2^{2^{(m+h\ell-1)/2}} \end{aligned}$$

for every $h \in \mathbb{N}$. From

$$\zeta_0 \cdot 2^{2^{(n+h\ell)/2}} = \delta_0 + \xi_0 \cdot 2^{2^{(m+h\ell-1)/2}}$$

it follows that $(n + h\ell)/2 = (m + h\ell - 1)/2$ and, thus, $m = n + 1$.

Likewise, we get

$$(\overrightarrow{\boldsymbol{\zeta}} \otimes \mathcal{S}^n(s_1))(h\ell + 1) = \zeta_1 \cdot t_2((n + h\ell)/2)$$
$$(\boldsymbol{\delta} \oplus (\overrightarrow{\boldsymbol{\xi}} \otimes \mathcal{S}^m(s_2)))(h\ell + 1) = \delta_1 + \xi_1 \cdot t_2((m + h\ell + 1)/2)$$

for every $h \in \mathbb{N}$. It follows that $(n + h\ell)/2 = (m + h\ell + 1)/2$ and, thus, $m = n - 1$.

We have reached a contradiction: $n - 1 = m = n + 1$, and, hence, our assumption must have been wrong. This concludes the proof. □

Acknowledgements We thank the editors for inviting us to contribute to this special issue, and the referees for their scrutiny of our paper and many useful suggestions. We thank Hans Zantema for fruitful discussions about the topic.

References

1. Allouche, J.-P., Shallit, J.: Automatic Sequences: Theory, Applications, Generalizations. Cambridge University Press, New York (2003)
2. Belov, A.: Some algebraic properties of machine poset of infinite words. ITA **42**(3), 451–466 (2008)
3. Berstel, J., Boasson, L., Carton, O., Petazzoni, B., Pin, J.-É.: Operations preserving recognizable languages. In: Fundamentals of Computation Theory, volume 2751 of LNCS, pp. 343–354. Springer, Berlin (2003)
4. Berstel, J., Boasson, L., Carton, O., Petazzoni, B., Pin, J.-É.: Operations preserving regular languages. Theoretical Computer Science **354**(3), 405–420 (2006)
5. Bosma, W., Zantema, H.: Ordering sequences by permutation transducers. Indagationes Mathematicae **28**(1), 38–54 (2017)
6. Dekking, F.M.: Iteration of maps by an automaton. Discrete Mathematics **126**(1–3), 81–86 (1994)
7. Endrullis, J.: Termination and Productivity. PhD thesis, Vrije Universiteit Amsterdam (2010)
8. Endrullis, J., de Vrijer, R.C., Waldmann, J.: Local termination: theory and practice. Log. Methods Comput. Sci. **6**(3), (2010)
9. Endrullis, J., Geuvers, H., Simonsen, J.G., Zantema, H.: Levels of undecidability in rewriting. Inf. Comput. **209**(2), 227–245 (2011)

10. Endrullis, J., Grabmayer, C., Hendriks, D.: Regularity preserving but not reflecting encodings. In: Proceedings of the Symposium on Logic in Computer Science (LICS 2015), pp 535–546. IEEE Computer Society (2015)
11. Endrullis, J., Grabmayer, C., Hendriks, D., Klop, J.W., de Vrijer, R.C.: Proving infinitary normalization. In: Proceedings of the Conference on Types for Proofs and Programs (TYPES 2008), volume 5497 of LNCS, pp. 64–82. Springer, Berlin (2008)
12. Endrullis, J., Grabmayer, C., Hendriks, D., Zantema, H.: The degree of squares is an atom. In: Proceedings of the Conference on Combinatorics on Words (WORDS 2015), volume 9304 of LNCS, pp. 109–121. Springer, Berlin (2015)
13. Endrullis, J., Hendriks, D.: Transforming outermost into context-sensitive rewriting. Log. Methods Comput. Sci. **6**(2), (2010)
14. Endrullis, J., Hendriks, D.: Lazy productivity via termination. Theor. Comput. Sci. **412**(28), 3203–3225 (2011)
15. Endrullis, J., Hendriks, D., Bakhshi, R.: On the complexity of equivalence of specifications of infinite objects. In: Proceedings of the International Conference on Functional Programming (ICFP 2012), pp, 153–164. ACM (2012)
16. Endrullis, J., Hendriks, D., Bakhshi, R., Rosu, G.: On the complexity of stream equality. J. Funct. Program. **24**(2–3), 166–217 (2014)
17. Endrullis, J., Hendriks, D., Klop, J.W.: Degrees of streams. J. Integers **11B**(A6):1–40 (2011). Proceedings of the Leiden Numeration Conference 2010
18. Endrullis, J., Hendriks, D., Klop, J.W.: Highlights in infinitary rewriting and lambda calculus. Theor. Comput. Sci. **464**, 48–71 (2012)
19. Endrullis, J., Karhumäki, J., Klop, J.W., Saarela, A.: Degrees of infinite words, polynomials and atoms. In: Proceedings of the Conference on Developments in Language Theory (DLT 2016), volume 9840 of LNCS, pp. 164–176. Springer, Berlin (2016)
20. Endrullis, J., Karhumäki, J., Klop, J.W., Saarela, A.: Degrees of infinite words, polynomials and atoms. Int. J. Found. Comput. Sci. **29**(5), 825–843 (2018)
21. Endrullis, J., Klop, J.W.: De Bruijn's weak diamond property revisited. Indagationes Mathematicae **24**(4), 1050–1072 (2013). In memory of N.G. (Dick) de Bruijn (1918-2012)
22. Endrullis, J., Klop, J.W., Overbeek, R.: Decreasing diagrams with two labels are complete for confluence of countable systems. In: Proceedings of the Conference on Formal Structures for Computation and Deduction (FSCD 2018), volume 108 of LIPIcs, pp. 14:1–14:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2018)
23. Endrullis, J., Klop, J.W., Saarela, A., Whiteland, M.A.: Degrees of transducibility. In: Proceedings of the Conference on Combinatorics on Words (WORDS 2015), volume 9304 of LNCS, pp. 1–13. Springer, Berlin (2015)
24. Endrullis, J., Polonsky, A.: Infinitary rewriting coinductively. In: Proceedings of the Conference on Types for Proofs and Programs (TYPES 2012), volume 19 of LIPIcs, pp. 16–27. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2011)
25. Endrullis, J., Zantema, H.: Proving non-termination by finite automata. In: Proceedings of the Conference on Rewriting Techniques and Applications (RTA 2015), volume 36 of LIPIcs, pp. 160–176. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2015)
26. Jacobs, K.: Invitation to Mathematics. Princeton University Press, Princeton (1992)
27. Kleene, S.C., Post, E.L.: The upper semi-lattice of degrees of recursive unsolvability. Ann. Math. **59**(3), 379–407 (1954)
28. Klop, J.W.: Term rewriting systems. In: Handbook of Logic in Computer Science, volume II, pp. 1–116. Oxford University Press, Oxford (1992)
29. Klop, J.W., de Vrijer, R.C.: Infinitary normalization. In: Artemov, S., Barringer, H., d'Avila Garcez, A.S., Lamb, L.C., Woods, J. (eds.) We Will Show Them: Essays in Honour of Dov Gabbay, volume 2, pp. 169–192. College Publications (2005)
30. Odifreddi, P.: Classical Recursion Theory. Studies in Logic and the Foundations of Mathematics. North-Holland, Amsterdam (1999)
31. Rayna, G.: Degrees of finite-state transformability. Inf. Control **24**(2), 144–154 (1974)
32. Sakarovitch, J.: Elements of Automata Theory. Cambridge University Press (2003)
33. Shoenfield, J.R.: Degrees of Unsolvability. North-Holland, Elsevier (1971)
34. Shore, R.A.: Conjectures and questions from Gerald Sacks's degrees of unsolvability. Arch. Math. Logic **36**(4–5), 233–253 (1997)
35. Soare, R.I.: Recursively enumerable sets and degrees. Bull. Am. Math. Soc. **84**(6), 1149–1181 (1978)
36. Spector, C.: On degrees of recursive unsolvability. Ann. Math. **64**(2), 581–592 (1956)

37. Sprunger, D., Tune, W., Endrullis, J., Moss, L.S.: Eigenvalues and transduction of morphic sequences. In: Proceedings of the Conference on Developments in Language Theory (DLT 2014), volume 8633 of LNCS, pp. 239–251. Springer, Berlin (2014)
38. Terese: Term Rewriting Systems, volume 55 of Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge (2003)
39. Zantema, H., Bosma, W.: Classifying non-periodic sequences by permutation transducers. In: Proceedings of the Conference on Developments in Language Theory (DLT 2017), volume 10396 of LNCS, pp. 365–377. Springer, Berlin (2017)
40. Zantema, H., Raffelsieper, M.: Proving productivity in infinite data structures. In: Proceedings of the 21st International Conference on Rewriting Techniques and Applications (RTA 2010), volume 6, pp. 401–416. Schloss Dagstuhl (2010)