

The Support of a Recognizable Series over a Zero-sum Free, Commutative Semiring is Recognizable[†]

Daniel Kirsten[‡]

Abstract

We show that the support of a recognizable series over a zero-sum free, commutative semiring is a recognizable language. We also give a sufficient and necessary condition for the existence of an effective transformation of a weighted automaton recognizing a series S over a zero-sum free, commutative semiring into an automaton recognizing the support of S .

Keywords: weighted automata, recognizable series, support

1 Introduction

One stream in the rich theory of formal power series deals with connections to formal languages. To each formal power series, one associates a certain language, called the support, which consists of all words which are not mapped to zero.

It is well-known that the support of a recognizable series is not necessarily a recognizable language. However, for large classes of semirings, it is known that the support of a recognizable series is always recognizable, see [3, 5, 9] for recent overviews. These classes include all positive semirings (semirings which are both zero-divisor free and zero-sum free), all finite, and more generally, all locally finite semirings.

WANG introduced the notion of a quasi-positive semiring (that is, for every $k \in \mathbb{K} \setminus \{0\}$, $\ell \in \mathbb{K}$, $n \in \mathbb{N}$, we have $k^n + \ell \neq 0$), and showed that the support of a recognizable series over a commutative, quasi-positive semiring is always a recognizable language [11]. Every quasi-positive semiring is zero-sum free by definition.

In 2008, MANFRED DROSTE raised the question whether WANG's result holds for commutative, zero-sum-free semirings. In the present paper, we answer this

[†]The results were achieved in 2008 when the author was employed in MANFRED DROSTE's group at Leipzig University. An extended abstract was presented at DLT'09 [6].

[‡]Humboldt-Universität zu Berlin, Institut für Informatik, Unter den Linden 6, D-10099 Berlin, Germany

question positively (see Theorem 3.1(1), below). The proof relies on DICKSON's lemma.

Further, we investigate under which assumptions we can effectively transform a weighted automaton recognizing a series S over a zero-sum free, commutative semiring into an automaton recognizing the support of S . For this, we introduce the zero generation problem (see Sect. 3) and show that the decidability of the zero generation problem is a sufficient and necessary condition for the existence of such an effective transformation. Surprisingly, the computability of the semiring operations is not related to the effectivity of the transformation.

The paper is organized as follows: in Sect. 2, we deal with some preliminaries. In Sect. 3, we present known results and the contribution of the paper. To keep Sect. 3 as a succinct survey, the main proofs are shifted to Sect. 4.

2 Preliminaries

2.1 Notations

Let $\mathbb{N} = \{0, 1, \dots\}$.

Let $n \in \mathbb{N}$. Given a tuple $\bar{x} \in \mathbb{N}^n$, we denote by x_i the i -th component of \bar{x} for $i \in \{1, \dots, n\}$. Given two tuples $\bar{x}, \bar{y} \in \mathbb{N}^n$, we write $\bar{x} \leq \bar{y}$ if $x_i \leq y_i$ for every $i \in \{1, \dots, n\}$. If $\bar{x} \leq \bar{y}$ and $x_i < y_i$ for some $i \in \{1, \dots, n\}$, then we write $\bar{x} < \bar{y}$.

Given a subset $M \subseteq \mathbb{N}^n$, we denote by $\text{Min}(M)$ the set of all minimal tuples of M , that is, $\text{Min}(M) = \{\bar{x} \in M \mid \text{for every } \bar{y} \in M, \bar{y} \leq \bar{x} \text{ implies } \bar{x} = \bar{y}\}$.

The following lemma is well-known in combinatorics, order theory, and commutative algebra. We include its proof for the convenience of the reader.

Lemma 2.1 (DICKSON's lemma). *For every $M \subseteq \mathbb{N}^n$, the set $\text{Min}(M)$ is finite.*

Proof. For $n = 1$, the claim is obvious.

Choose some $n \in \mathbb{N}$, and assume by induction that the claim holds for all subsets of \mathbb{N}^n . We show the claim for an arbitrary $M \subseteq \mathbb{N}^{n+1}$.

For $z \in \mathbb{N}$, let

$$M_z := \{(x_1, \dots, x_n) \mid (x_1, \dots, x_n, z) \in \text{Min}(M)\}.$$

Clearly, $\text{Min}(M_z) = M_z$, and hence, M_z is finite by induction. Let

$$M_{\mathbb{N}} := \bigcup_{z \in \mathbb{N}} M_z.$$

By induction $\text{Min}(M_{\mathbb{N}})$ is finite, and thus, there is some $z' \in \mathbb{N}$ such that

$$\text{Min}(M_{\mathbb{N}}) \subseteq \bigcup_{z \leq z'} M_z.$$

Now, we show the claim by showing that

$$\text{Min}(M) \subseteq \bigcup_{z \leq z'} M_z \times \{z\}, \tag{1}$$

i.e., $\text{Min}(M)$ is included in a finite union of finite sets. The notation $M_z \times \{z\}$ means to adjoin z as $(n+1)$ -st component to each n -tuple in M_z .

Choose any $\bar{x} \in \text{Min}(M)$. Clearly, $(x_1, \dots, x_n) \in M_{x_{n+1}} \subseteq M_{\mathbb{N}}$. There is some $\bar{y} \in \text{Min}(M_{\mathbb{N}})$ satisfying $\bar{y} \leq (x_1, \dots, x_n)$. There is some $z \leq z'$ such that $\bar{y} \in M_z$. If $z < x_{n+1}$ then $(y_1, \dots, y_n, z) < \bar{x}$ contradicts $\bar{x} \in \text{Min}(M)$. Hence, $x_{n+1} \leq z \leq z'$. Consequently, \bar{x} belongs to the right hand side of (1). \square

Let Σ be a finite alphabet. We denote the empty word by ε . We denote by $|w|$ the length of a word $w \in \Sigma^*$. For every $w \in \Sigma^*$, $a \in \Sigma$, let $|w|_a$ be the number of occurrences of the letter a in w .

A monoid $(\mathbb{M}, \cdot, 1)$ consists of a set \mathbb{M} together with a binary associative operation \cdot and an identity 1 .

We call a monoid $(\mathbb{M}, \cdot, 1)$ *commutative* if $kl = lk$ for every $k, l \in \mathbb{M}$.

We call $0 \in \mathbb{M}$ a *zero*, if $k0 = 0k = 0$ for every $k \in \mathbb{M}$.

Given a monoid \mathbb{M} , $m \in \mathbb{N}$, and $s_1, \dots, s_m \in \mathbb{M}$, we denote by $\langle s_1, \dots, s_m \rangle$ the submonoid of \mathbb{M} generated by s_1, \dots, s_m , that is, the smallest monoid $\mathbb{M}' \subseteq \mathbb{M}$ satisfying $s_1, \dots, s_m \in \mathbb{M}'$.

Given a monoid \mathbb{M} , an $s \in \mathbb{M}$, and a submonoid $\mathbb{M}' \subseteq \mathbb{M}$, we denote by $s \cdot \mathbb{M}'$ the set $\{s \cdot s' \mid s' \in \mathbb{M}'\}$.

A *semiring* $(\mathbb{K}, +, \cdot, 0, 1)$ consists of a set \mathbb{K} together with two binary operations $+$ and \cdot such that $(\mathbb{K}, +, 0)$ is a commutative monoid, $(\mathbb{K}, \cdot, 1)$ is a monoid with zero 0 , and $(\mathbb{K}, \cdot, 1)$ distributes over $(\mathbb{K}, +, 0)$.

We call a semiring $(\mathbb{K}, +, \cdot, 0, 1)$ *commutative* if $(\mathbb{K}, \cdot, 1)$ is a commutative monoid. We call \mathbb{K} *zero-divisor free* if for every $k, l \in \mathbb{K} \setminus \{0\}$, we have $kl \neq 0$. We call \mathbb{K} *zero-sum free* if for every $k, l \in \mathbb{K} \setminus \{0\}$, we have $k + l \neq 0$. Semirings which are both zero-divisor free and zero-sum free are called *positive* semirings.

We call \mathbb{K} *locally finite* if for every finite subset $C \subseteq \mathbb{K}$, there is a finite semiring \mathbb{K}' satisfying $C \subseteq \mathbb{K}' \subseteq \mathbb{K}$.

2.2 Weighted Finite Automata

We recall some notions on (weighted) automata and recommend [1, 2, 4, 7, 8, 10] for overviews.

Let $(\mathbb{K}, +, \cdot, 0, 1)$ be a semiring. Mappings from Σ^* to \mathbb{K} are often called *series*. We denote the class of all series from Σ^* to \mathbb{K} by $\mathbb{K}\langle\langle\Sigma^*\rangle\rangle$.

A *weighted finite automaton* (for short *WFA*) over \mathbb{K} is a tuple $[Q, E, \lambda, \varrho]$, where

- Q is a non-empty, finite set of *states*,
- E is a finite subset of $Q \times \Sigma \times \mathbb{K} \times Q$, and
- $\lambda, \varrho : Q \rightarrow \mathbb{K}$.

We call the tuples in E *transitions*. For every $q \in Q$, we call $\lambda(q)$ resp. $\varrho(q)$ the *initial weight* resp. *accepting weight* of q . We call states $q \in Q$ which satisfy $\lambda(q) \neq 0$ (resp. $\varrho(q) \neq 0$) *initial* (resp. *accepting*) states.

Let $\mathcal{A} = [Q, E, \lambda, \varrho]$ be a WFA. Let $n \geq 1$. A path π of length n is a sequence

$$(q_0, a_1, s_1, q_1) (q_1, a_2, s_2, q_2) \dots (q_{n-1}, a_n, s_n, q_n)$$

of transitions in E . We call the word $a_1 \dots a_n$ the *label* of π . We define $\sigma(\pi) = \lambda(q_0) \cdot s_1 \cdot s_2 \cdot \dots \cdot s_n \cdot \varrho(q_n)$, the *weight* of π . For every state $q \in Q$, we assume some path from q to q which is labeled with ε and weighted with 1.

For every $p, q \in Q$ and every $w \in \Sigma^*$, we denote by $p \overset{w}{\rightsquigarrow} q$ the set of all paths with label w which start at p and end at q . Then, \mathcal{A} defines a series $|\mathcal{A}| : \Sigma^* \rightarrow \mathbb{K}$ by

$$|\mathcal{A}|(w) = \sum_{p, q \in Q, \pi \in p \overset{w}{\rightsquigarrow} q} \sigma(\pi)$$

for every $w \in \Sigma^*$.

We call a series $S : \Sigma^* \rightarrow \mathbb{K}$ *recognizable* if $S = |\mathcal{A}|$ for some WFA \mathcal{A} .

We define the *support* of a series $S : \Sigma^* \rightarrow \mathbb{K}$ as

$$\text{supp}(S) = \{w \in \Sigma^* \mid S(w) \neq 0\}.$$

An (*unweighted*) *automaton* is a tuple $\mathcal{A} = [Q, E, I, F]$, where Q is a finite set, $E \subseteq Q \times \Sigma \times Q$, $I \subseteq Q$, and $F \subseteq Q$.

Let $\mathcal{A} = [Q, E, \lambda, \varrho]$ be an automaton. Let $n \geq 1$. A path π of length n is a sequence

$$(q_0, a_1, q_1) (q_1, a_2, q_2) \dots (q_{n-1}, a_n, q_n)$$

of transitions in E . As above, we call $a_1 \dots a_n$ the *label* of π . We call π *successful*, if $q_0 \in I$ and $q_n \in F$. We denote by $L(\mathcal{A})$ the language of \mathcal{A} , that is, the language consisting of all labels of successful paths.

3 Overview, Main Results, and Discussion

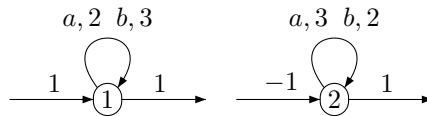
The supports of recognizable series are well-studied objects, see [3, 9] for recent overviews.

It is well known that there are recognizable series S such that $\text{supp}(S)$ is not a recognizable language.

Example 3.1. A folklore example is the series S over the semiring of the integers $(\mathbb{Z}, +, \cdot, 0, 1)$ defined by $S(w) = 2^{|w|_a} 3^{|w|_b} - 3^{|w|_a} 2^{|w|_b}$. For every $w \in \Sigma^*$, we have $S(w) = 0$ iff $|w|_a = |w|_b$. Hence,

$$\text{supp}(S) = \{w \in \Sigma^* \mid |w|_a \neq |w|_b\}$$

which is not a recognizable language. Nevertheless, S is a recognizable series: just consider the WFA given below.



However, for large classes of semirings, the support of a recognizable series is always a recognizable language. It is well known that these classes include all positive semirings, all finite and moreover even all locally finite semirings [3, 5, 9].

Moreover, WANG [11] defined the notion of a quasi-positive semiring: a semiring \mathbb{K} is called *quasi-positive* if for every $k \in \mathbb{K} \setminus \{0\}$, $\ell \in \mathbb{K}$, $n \in \mathbb{N}$, we have $k^n + \ell \neq 0$. Every positive semiring is quasi-positive, and every quasi-positive semiring is zero-sum free. There are quasi-positive semirings which are not positive. Just let $\mathbb{K} = \mathbb{N} \times \mathbb{N}$ equipped with componentwise addition and multiplication of integers.

Moreover, there are zero-sum free semirings which are not quasi-positive.

Example 3.2. Let \mathbb{K} be the semiring of (2×2) -matrices over the non-negative rational numbers $(\mathbb{Q}_+, +, \cdot, 0, 1)$ and let

$$k = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad \ell = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Clearly, $k^2 + \ell$ yields the zero matrix, and hence, \mathbb{K} is not quasi-positive but zero-sum-free.

In the context of our main result, it raises the question for a commutative, zero-sum free semiring which is not quasi-positive. Indeed,¹ let \mathbb{K}' be the subset of \mathbb{K} consisting of all matrices of the form

$$\begin{pmatrix} x & y \\ 0 & x \end{pmatrix} \quad \text{for} \quad x, y \in \mathbb{Q}_+.$$

It is easy to verify that \mathbb{K}' is a commutative subsemiring of \mathbb{K} . It is zero-sum-free, and since $k, \ell \in \mathbb{K}'$, it is not quasi-positive.

WANG showed that for every recognizable series S over a commutative, quasi-positive semiring, $\text{supp}(S)$ is recognizable [11]. In 2008, MANFRED DROSTE raised the question whether WANG's result holds for commutative, zero-sum-free semirings in a lecture script on weighted automata theory. In the present paper, we answer this question positively (see Theorem 3.1(1), below). Our approach is quite different from WANG's paper [11], since WANG was mainly interested in other but related questions and achieved his result as a byproduct.

One key observation is that for zero-sum-free semirings, a word w belongs to the support of the series of some WFA iff the WFA admits at least one path for w with a non-zero weight. In contrast to Example 3.1, it cannot happen that the weights of all paths for w are summarized to 0.

Further, we examine under which assumptions we can effectively construct an automaton recognizing $\text{supp}(S)$ from a WFA recognizing S . Surprisingly, the computability of $+$ or \cdot is not related to the effectivity of the construction. To achieve an effective construction, we introduce the *zero generation problem* (for short *ZGP*):

Let \mathbb{M} be a monoid with a zero. An instance of the ZGP consists of two integers $m, m' \in \mathbb{N}$ and $s_1, \dots, s_m, s'_1, \dots, s'_{m'} \in \mathbb{M}$. The ZGP means to decide whether

¹The semiring \mathbb{K}' was provided by an anonymous referee.

$0 \in s_1 \cdots s_m \cdot \langle s'_1, \dots, s'_{m'} \rangle$, i.e., whether there exists some $s \in \langle s'_1, \dots, s'_{m'} \rangle$ such that the product $s_1 \cdots s_m \cdot s$ yields zero. The presentation of the ZGP seems to be circumstantial, but we want to avoid using the computability of the product in \mathbb{M} .

Note that the integers m and m' in the ZGP are allowed to be 0. Consequently, the problem to decide whether for given $m \in \mathbb{N}$ and $s_1, \dots, s_m \in \mathbb{M}$, we have $s_1 \cdots s_m = 0$ is a particular case of the ZGP.

We can show that the decidability of the ZGP of the monoid $(\mathbb{K}, \cdot, 1)$ is a sufficient and necessary condition for the effectivity of the construction of the automaton recognizing the support of some recognizable series over a semiring \mathbb{K} .

To sum up:

Theorem 3.1. *Let Σ be an alphabet and $(\mathbb{K}, +, \cdot, 0, 1)$ be a zero-sum free, commutative semiring.*

1. *For every recognizable series $S \in \mathbb{K}\langle\langle \Sigma^* \rangle\rangle$, $\text{supp}(S)$ is a recognizable language.*
2. *Assume $|\Sigma| \geq 2$. Given a WFA \mathcal{A} over \mathbb{K} , we can effectively construct an automaton which recognizes $\text{supp}(|\mathcal{A}|)$ iff $(\mathbb{K}, \cdot, 1)$ has a decidable ZGP.*

Clearly, the construction in (2) is also effective for $|\Sigma| = 1$. But if $|\Sigma| = 1$ we cannot show that the decidability of the ZGP is a necessary condition.

Unfortunately, we cannot give any reasonable upper bound in the construction in Theorem 3.1(2). Given a WFA \mathcal{A} over a zero-sum free, commutative semiring \mathbb{K} , the number of states of an automaton recognizing $\text{supp}(|\mathcal{A}|)$ does not only depend on the number of states of \mathcal{A} and the weights in \mathcal{A} , but also it highly depends on structural properties of the semiring \mathbb{K} . The construction of the automaton recognizing $\text{supp}(|\mathcal{A}|)$ in the proof of Theorem 3.1(2) involves a certain bound which is computed in a brute search using some algorithm for the ZGP. The existence of this bound is guaranteed by DICKSON's lemma (Lemma 2.1).

4 The Main Proof

4.1 Dickson's Lemma and Computability

Throughout this section, let $(\mathbb{M}, \cdot, 1)$ be a commutative monoid with a zero 0 and let $C = (c_1, \dots, c_n) \in \mathbb{M}^n$ for some $n \in \mathbb{N}$.

The homomorphism $\llbracket \cdot \rrbracket : (\mathbb{N}^n, +, (0, \dots, 0)) \rightarrow (\mathbb{M}, \cdot, 1)$ defined by

$$\llbracket \bar{x} \rrbracket = c_1^{x_1} \cdots c_n^{x_n}$$

for every $\bar{x} = (x_1, \dots, x_n) \in \mathbb{N}^n$ plays a central role in the entire construction. Let us remark that the commutativity of \mathbb{M} is crucial for the fact that $\llbracket \cdot \rrbracket$ is a homomorphism which will be of crucial importance, e.g., in the proof of Lemma 4.1, below.

We are interested in the set of all $\bar{x} \in \mathbb{N}^n$ satisfying $\llbracket \bar{x} \rrbracket = 0$, i.e., we are interested in the set $\llbracket 0 \rrbracket^{-1}$.

Given $\bar{x} \in \llbracket 0 \rrbracket^{-1}$ and $\bar{y} \in \mathbb{N}^n$ satisfying $\bar{x} \leq \bar{y}$, we have $\bar{y} \in \llbracket 0 \rrbracket^{-1}$.

By Lemma 2.1, the set $\text{Min}(\llbracket 0 \rrbracket^{-1})$ is finite. We denote by $\text{dg}(C)$ the *degree* of C which is defined as the least non-negative integer such that $\text{Min}(\llbracket 0 \rrbracket^{-1})$ is a subset of $\{0, \dots, \text{dg}(C)\}^n$.

Example 4.1. Let us consider a commutative monoid which admits large degrees. Let $\mathbb{M} := \{q \in \mathbb{Q} \mid 0 \leq q \leq 1\}$. We define an operation \star on \mathbb{M} by setting $p \star q := \min\{p + q, 1\}$ for $p, q \in \mathbb{M}$. Clearly, $(\mathbb{M}, \star, 0)$ is a commutative monoid with zero 1.

Now, let $n \in \mathbb{N}$ and $c_i \in \mathbb{M}$ for $i \in \{1, \dots, n\}$. If $c_i \neq 0$, then

$$\left(0, \dots, 0, \underbrace{\left\lceil \frac{1}{c_i} \right\rceil}_{i\text{th position}}, 0, \dots, 0\right) \in \text{Min}\llbracket 1 \rrbracket^{-1},$$

where $\left\lceil \frac{1}{c_i} \right\rceil$ denotes the least integer larger than or equal to $\frac{1}{c_i}$. Consequently, $\text{dg}(C) \geq \frac{1}{c_i}$.

Given $\bar{x} \in \mathbb{N}^n$ and $z \in \mathbb{N}$, we denote by $[\bar{x}]_z$ the tuple defined by $([\bar{x}]_z)_i = \min\{x_i, z\}$ for every $i \in \{1, \dots, n\}$.

Lemma 4.1. *For every $\bar{x} \in \mathbb{N}^n$, we have $\llbracket \bar{x} \rrbracket = 0$ iff $\llbracket [\bar{x}]_{\text{dg}(C)} \rrbracket = 0$.*

Proof. We have “ \Leftarrow ”, since $\bar{x} \geq [\bar{x}]_{\text{dg}(C)}$.

We show “ \Rightarrow ”. Since $\bar{x} \in \llbracket 0 \rrbracket^{-1}$, there is a $\bar{y} \in \text{Min}(\llbracket 0 \rrbracket^{-1})$ satisfying $\bar{y} \leq \bar{x}$. Let $i \in \{1, \dots, n\}$. If $x_i \leq \text{dg}(C)$, then $y_i \leq x_i = ([\bar{x}]_{\text{dg}(C)})_i$. If $x_i > \text{dg}(C)$, then $y_i \leq \text{dg}(C) = ([\bar{x}]_{\text{dg}(C)})_i$ by the definitions of $\text{dg}(C)$ and $[\bar{x}]_{\text{dg}(C)}$. Consequently, $\bar{y} \leq [\bar{x}]_{\text{dg}(C)}$, and hence, $[\bar{x}]_{\text{dg}(C)} \in \llbracket 0 \rrbracket^{-1}$. \square

For the effectivity of our construction of the support automaton, it is very important to compute $\text{dg}(C)$ from a given tuple C .

Lemma 4.2. *If the ZGP is decidable in \mathbb{M} , then we can effectively compute $\text{dg}(C)$ from C .*

Proof. It suffices to show that for given $n \in \mathbb{N}$, $C = (c_1, \dots, c_n) \in \mathbb{M}^n$, and $z \in \mathbb{N}$, we can decide whether $z < \text{dg}(C)$. The algorithm can then check for increasing $z \in \{0, 1, 2, \dots\}$ whether $z < \text{dg}(C)$, and put out the least z which does not satisfy $z < \text{dg}(C)$.

So assume n, C, z as above. We want to show that $z < \text{dg}(C)$ iff there exists a tuple $\bar{x} \in \{0, \dots, z\}^n$ which satisfies the following properties:

1. We have $x_i = z$ for some $i \in \{1, \dots, n\}$.
2. We have $\llbracket \bar{x} \rrbracket \neq 0$. Given C and \bar{x} , it is decidable whether $\llbracket \bar{x} \rrbracket \neq 0$ by the decidability of the ZGP.
3. There is some $\bar{y} \in \mathbb{N}^n$ such that $\bar{x} = [\bar{y}]_z$ and $\llbracket \bar{y} \rrbracket = 0$.

Given C and \bar{x} , this condition is decidable as follows: Let $m = \sum_{i=1}^n x_i$. Let s_1, \dots, s_m be the list over \mathbb{M} constructed by putting x_1 times c_1 , x_2 times c_2 , \dots , and x_n times c_n . We have $s_1 \cdots s_m = \llbracket \bar{x} \rrbracket$.

Let $m' \geq 1$ and $s'_1, \dots, s'_{m'} \in \mathbb{M}$ be a list of the c_i 's for the $i \in \{1, \dots, n\}$ satisfying $x_i = z$.

Clearly, there exists some $\bar{y} \in \mathbb{N}^n$ such that $\bar{x} = \lfloor \bar{y} \rfloor_z$ and $\llbracket \bar{y} \rrbracket = 0$ iff $0 \in s_1 \cdots s_m \cdot \langle s'_1, \dots, s'_{m'} \rangle$. The latter condition is decidable.

Assume $z < \text{dg}(C)$. Choose a $\bar{y} \in \text{Min}(\llbracket 0 \rrbracket^{-1})$ such that at least one entry of \bar{y} equals $\text{dg}(C)$. Let $\bar{x} = \lfloor \bar{y} \rfloor_z$. Obviously, \bar{x} satisfies (1) and (3). Since $\bar{x} < \bar{y}$, we have $\bar{x} \notin \llbracket 0 \rrbracket^{-1}$, and hence, \bar{x} satisfies (2).

Assume $z \geq \text{dg}(C)$. Let $\bar{x}, \bar{y} \in \mathbb{N}^n$ such that (1) and (3) are satisfied. From Lemma 4.1, it follows $\llbracket \lfloor \bar{y} \rfloor_{\text{dg}(C)} \rrbracket = 0$. Since $\text{dg}(C) \leq z$, we have $\lfloor \bar{y} \rfloor_{\text{dg}(C)} \leq \lfloor \bar{y} \rfloor_z = \bar{x}$, and hence, $\llbracket \bar{x} \rrbracket = 0$, i.e., \bar{x} does not satisfy (2).

An algorithm to decide whether $z < \text{dg}(C)$ can check by brute force whether there is an $\bar{x} \in \{0, \dots, z\}^n$ which satisfies (1), (2), and (3). \square

4.2 The Construction of a Support Automaton

Proof of Theorem 3.1. In the first part of the proof we prove (1) and “ \Leftarrow ” in (2).

Let S be the series computed by a WFA $\mathcal{A} = [Q, E, \lambda, \varrho]$.

Let C be a sequence (without repetition) of all weights occurring in \mathcal{A} . That is, let $n \in \mathbb{N}$ and $C = (c_1, \dots, c_n) \in \mathbb{K}^n$ such that:

- For every $i \in \{1, \dots, n\}$, there is a transition $(p, a, c_i, q) \in E$ or there is a $q \in Q$ satisfying $\lambda(q) = c_i$ or $\varrho(q) = c_i$.
- For every $(p, a, s, q) \in E$, there is exactly one $i \in \{1, \dots, n\}$ satisfying $c_i = s$.
- For every $q \in Q$, there is exactly one $i \in \{1, \dots, n\}$ satisfying $\lambda(q) = c_i$, and there is exactly one $i \in \{1, \dots, n\}$ satisfying $\varrho(q) = c_i$.

We construct an (unweighted) automaton \mathcal{A}_s . We will use $\text{dg}(C)$ in a crucial way. If the ZGP is decidable, we can effectively compute $\text{dg}(C)$ by Lemma 4.2 and then, our construction is effective.

The state set of \mathcal{A}_s is $Q_s = \{0, \dots, \text{dg}(C)\}^n \times Q$.

A state $(\bar{x}, q) \in Q_s$ is an initial state iff there exists some $i \in \{1, \dots, n\}$ such that

- $x_i = 1$, $\lambda(q) = c_i$, and
- for every $j \in \{1, \dots, n\}$, $j \neq i$, we have $x_j = 0$.

Consequently, $\llbracket \bar{x} \rrbracket = c_i = \lambda(q)$. We denote the set of initial states by I_s .

We could also define the set of initial states by $I'_s = \{(\bar{x}, q) \in Q_s \mid \llbracket \bar{x} \rrbracket = \lambda(q)\}$ which is a superset of I_s . One can easily construct examples for which $I_s \subsetneq I'_s$. Just consider the case that for some $(\bar{x}, q) \in Q_s$, we have $x_1 = x_2 = 1$, $x_3 = \dots = x_n = 0$

and $c_1c_2 = \lambda(q)$. Our construction below remains correct even if we use I'_s instead of I_s . However, the definition of I'_s involves the decision problem $\llbracket \bar{x} \rrbracket = \lambda(q)$ which we want to avoid to get an effective construction.

We define a partial mapping $\oplus : \{0, \dots, \text{dg}(C)\}^n \times \mathbb{K} \dashrightarrow \{0, \dots, \text{dg}(C)\}^n$. The key idea behind \oplus is that given $m \in \mathbb{N}$, $s_1, \dots, s_m \in \mathbb{K}$, the operation

$$(\dots((\bar{x} \oplus s_1) \oplus s_2) \dots \oplus s_m)$$

counts (up to $\text{dg}(C)$) the number of occurrences of the c_i 's in the sequence s_1, \dots, s_m .

Let $\bar{x} \in \{0, \dots, \text{dg}(C)\}^n$ and $s \in \mathbb{K}$. We define $\bar{x} \oplus s$ iff there is some $i \in \{1, \dots, n\}$ satisfying $c_i = s$. Let $\bar{y} \in \{0, \dots, \text{dg}(C)\}^n$ be defined by

$$y_j = \begin{cases} x_j + 1 & \text{if } j = i \\ x_j & \text{if } j \neq i. \end{cases}$$

We define $\bar{x} \oplus s = \lfloor \bar{y} \rfloor_{\text{dg}(C)}$.

A state $(\bar{x}, q) \in Q_s$ is an accepting state iff $\llbracket \bar{x} \oplus \varrho(q) \rrbracket \neq 0$. Using the decidability of the ZGP, we can decide whether (\bar{x}, q) is an accepting state. We denote the set of accepting states by F_s .

Let $(\bar{x}, p), (\bar{y}, q) \in Q_s$ and $a \in \Sigma$. The triple $((\bar{x}, p), a, (\bar{y}, q))$ is a transition in E_s iff there exists a transition $(p, a, s, q) \in E$ satisfying $\bar{x} \oplus s = \bar{y}$. We say that $((\bar{x}, p), a, (\bar{y}, q))$ stems from $(p, a, s, q) \in E$.

Let $\mathcal{A}_s = [Q_s, E_s, I_s, F_s]$. We want to show $L(\mathcal{A}_s) = \text{supp}(S)$.

Let $w \in L(\mathcal{A}_s)$. There are $(\bar{x}_0, q_0) \in I_s$, $(\bar{x}_{|w|}, q_{|w|}) \in F_s$, and some path $\pi \in (\bar{x}_0, q_0) \overset{w}{\rightsquigarrow} (\bar{x}_{|w|}, q_{|w|})$ satisfying $\llbracket \bar{x}_{|w|} \oplus \varrho(q_{|w|}) \rrbracket \neq 0$.

We denote the states of π by $(\bar{x}_0, q_0), (\bar{x}_1, q_1), \dots, (\bar{x}_{|w|}, q_{|w|})$.

For $j \in \{1, \dots, |w|\}$, let $t_j \in E$ such that the j -th transition of π stems from t_j .

Clearly, $t_1 \dots t_{|w|} \in q_0 \overset{w}{\rightsquigarrow} q_{|w|}$ is a path in \mathcal{A} .

For every $j \in \{1, \dots, |w|\}$, let $s_j \in \mathbb{K}$ be the weight of t_j . For $j \in \{0, \dots, |w|\}$, let $\bar{y}_j \in \mathbb{N}^n$ be the tuple such that for every $i \in \{1, \dots, n\}$, $y_{j,i}$ is the number of occurrences of c_i among $\lambda(q_0), s_1, \dots, s_j$. In particular $\bar{y}_0 = \bar{x}_0$.

Let $\bar{y} \in \mathbb{N}^n$ such that for every $i \in \{1, \dots, n\}$, y_i is the number of occurrences of c_i among $\lambda(q_0), s_1, \dots, s_{|w|}, \varrho(q_{|w|})$. Clearly, $\llbracket \bar{y} \rrbracket$ is the weight of the path $t_1 \dots t_{|w|}$.

By a straightforward inductive argument, we can show that for every $j \in \{0, \dots, |w|\}$, $\bar{x}_j = \lfloor \bar{y}_j \rfloor_{\text{dg}(C)}$, and $\bar{x}_{|w|} \oplus \varrho(q_{|w|}) = \lfloor \bar{y} \rfloor_{\text{dg}(C)}$.

Since $(\bar{x}_{|w|}, q_{|w|}) \in F_s$, we have $\llbracket \bar{x}_{|w|} \oplus \varrho(q_{|w|}) \rrbracket \neq 0$, and hence, $\llbracket \lfloor \bar{y} \rfloor_{\text{dg}(C)} \rrbracket \neq 0$. By Lemma 4.1, we have $\llbracket \bar{y} \rrbracket \neq 0$, i.e., the weight of the path $t_1 \dots t_{|w|}$ is different from 0. Since \mathbb{K} is zero-sum-free, we have $w \in \text{supp}(|\mathcal{A}|)$.

Thus, we have shown $L(\mathcal{A}_s) \subseteq \text{supp}(|\mathcal{A}|)$. To show $L(\mathcal{A}_s) \supseteq \text{supp}(|\mathcal{A}|)$, we can proceed in the same way. We assume some $w \in \text{supp}(|\mathcal{A}|)$, some accepting path $t_1 \dots t_{|w|}$ with non-zero weight for w in \mathcal{A} . For $j \in \{1, \dots, |w|\}$, we denote $t_j = (q_{j-1}, a_j, s_j, q_j)$. Let $\bar{x}_0 = (0, \dots, 0) \oplus \lambda(q_0)$. For $j \in \{1, \dots, |w|\}$, let $\bar{x}_j = \bar{x}_{j-1} \oplus s_j$. We can argue as above to show that the transitions $((\bar{x}_{i-1}, q_{i-1}), a_j, (\bar{x}_i, q_i))$ form an accepting path for w in \mathcal{A}_s . To sum up, $L(\mathcal{A}_s) = \text{supp}(|\mathcal{A}|)$.

We have shown (1) and “ \Leftarrow ” in (2). It remains to show “ \Rightarrow ” in (2). Assume an instance of the ZGP, i.e., let $m, m' \in \mathbb{N}$ and $s_1, \dots, s_m, s'_1, \dots, s'_{m'} \in \mathbb{K}$.

Let $w_1, \dots, w_{m'} \in \Sigma^*$ be mutually distinct, non-empty words of equal length.²

We sketch the construction of a WFA \mathcal{A} . It has just one initial and one accepting state. The initial and accepting weights are 1. Let a be some letter from Σ . For now, there is exactly one path from the initial to the accepting state. This path is labeled with a^m . The transition weights along this path are s_1, \dots, s_m . For every $j \in \{1, \dots, m'\}$, we add a loop at the accepting state which is labeled with w_j . The first transition of the loop is weighted with s'_j , the remaining transitions of the loop are weighted with 1.

For every n and $i_1, \dots, i_n \in \{1, \dots, m'\}$, we have

$$|\mathcal{A}|(a^m w_{i_1} \dots w_{i_n}) = s_1 \cdots s_m \cdot s'_{i_1} \cdots s'_{i_n}.$$

Moreover, we have $\text{supp}(|\mathcal{A}|) = a^m \{w_1, \dots, w_{m'}\}^*$ iff $0 \notin s_1 \cdots s_m \cdot \langle s'_1 \cdots s'_{m'} \rangle$.

By the assumption of “ \Rightarrow ” in (2), we can effectively construct an automaton \mathcal{A}_s which recognizes $\text{supp}(|\mathcal{A}|)$. By checking $L(\mathcal{A}_s) = a^m \{w_1, \dots, w_{m'}\}^*$, we can check whether $\text{supp}(|\mathcal{A}|) = a^m \{w_1, \dots, w_{m'}\}^*$, i.e., whether $0 \notin s_1 \cdots s_m \cdot \langle s'_1 \cdots s'_{m'} \rangle$. \square

Acknowledgements

The author thanks the anonymous reviewers of the present paper and its extended abstract at DLT'09 [6]. The author greatly acknowledges the example of a commutative, zero-sum free semiring which is not quasi-positive provided by an anonymous referee shown in Example 3.2.

References

- [1] Berstel, J. *Transductions and Context-Free Languages*. B. G. Teubner, Stuttgart, 1979.
- [2] Berstel, J. and Reutenauer, C. *Rational Series and Their Languages*, volume 12 of *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, Berlin Heidelberg New York, 1984.
- [3] Berstel, J. and Reutenauer, C. *Noncommutative Rational Series With Applications*, volume 137 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, 2010.
- [4] Droste, M., Kuich, W., and Vogler, H., editors. *Handbook of Weighted Automata*. Monographs in Theoretical Computer Science. An EATCS Series. Springer-Verlag, 2009.

²At this point, we need $|\Sigma| > 1$.

- [5] Kirsten, D. An algebraic characterization of semirings for which the support of every recognizable series is recognizable. In Kráľovič, R. and Niwiński, D., editors, *MFCS'09 Proceedings*, volume 5734 of *LNCS*, pages 489–500. Springer-Verlag, Berlin, 2009.
- [6] Kirsten, D. The support of a recognizable series over a zero-sum free, commutative semiring is recognizable. In Diekert, V. and Nowotka, D., editors, *DLT'09 Proceedings*, volume 5583 of *LNCS*, pages 326–333. Springer-Verlag, Berlin, 2009.
- [7] Kuich, W. Semirings and formal power series. In Rozenberg, G. and Salomaa, A., editors, *Handbook of Formal Languages, Vol. 1, Word, Language, Grammar*, pages 609–677. Springer-Verlag, Berlin, 1997.
- [8] Reutenauer, C. A survey on noncommutative rational series. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 24:159–169, 1996.
- [9] Sakarovitch, J. Rational and recognisable power series. Chapter 4 in [4], 2009.
- [10] Salomaa, A. and Soittola, M. *Automata-Theoretic Aspects of Formal Power Series*. Texts and Monographs on Computer Science. Springer-Verlag, Berlin Heidelberg New York, 1978.
- [11] Wang, H. On rational series and rational languages. *Theoretical Computer Science*, 205(1-2):329–336, 1998.