Determinisability of Unary Weighted Automata over the Rational Numbers

Peter Kostolányi[1]

Department of Computer Science, Comenius University in Bratislava,
Mlynská dolina, 842 48 Bratislava, Slovakia

Abstract

Weighted finite automata over the field of rational numbers and unary alphabets are considered. The notion of a characteristic polynomial is introduced for such automata as a means to provide a decidable necessary and sufficient condition, under which a unary weighted automaton admits a deterministic, *i.e.*, sequential equivalent. The sequentiality problem for univariate rational series is thus proved to be decidable both over the rational numbers and over the integers, confirming a conjecture of S. Lombardy and J. Sakarovitch; its decidability over the nonnegative integers is observed as well. The decision algorithm proposed for these tasks is shown to run in polynomial time. A determinisation algorithm for determinisable unary weighted automata over the rational numbers is also described.

Keywords: Deterministic weighted automaton, Sequential weighted automaton, Reduced representation, Characteristic polynomial, Cyclotomic polynomial

1. Introduction

Formal power series in several noncommuting variables form a well-known generalisation of formal languages, allowing one to replace a *qualitative* property of membership of a word in a language by a *quantity* taken from a suitable domain such as a semiring. This corresponds on an effective level to a transition from the usual descriptive mechanisms for formal languages – such as automata, grammars, rational expressions, or MSO logics – to their weighted counterparts. Specifically, nondeterministic finite automata recognising rational languages are generalised to weighted finite automata, which realise rational formal power series. The reader is referred to [8, 18, 19, 41] for a general overview of the related theory.

One of the aspects involved in this generalisation is that the classical equivalence of nondeterministic and deterministic finite automata does not lift to the quantitative setting – there are weighted automata admitting no deterministic equivalents [34, 37]. Deterministic weighted automata – also called *sequential* [34] or *subsequential* [36] by some authors[2] – thus not only form a proper subclass of weighted automata, but the corresponding class of series, called *sequential* in this article, also forms a proper subclass of the class of rational series. As determinism is often a crucial requirement in practice, the problems of algorithmically deciding determinisability of a weighted automaton, and of actually finding a deterministic equivalent when possible, have both received significant attention in literature – they have been studied, in different forms and often under some additional restrictions, over various classes of semirings [1, 31, 32, 34, 36, 37] (where especially the tropical semirings are of special importance), as well as over strong bimonoids [15], including research pertaining to the theory of weighted tree automata [16, 23].

Email address: kostolanyi@fmph.uniba.sk (Peter Kostolányi)

[1]The author was supported by the grant VEGA 1/0601/20.

[2]The terminology is not consistent here, as the term "sequential" is used in a more restrictive sense by the authors who call deterministic automata subsequential. See S. Lombardy and J. Sakarovitch [34] for more details. We stick with the term "deterministic" in the context of automata, while we use the term "sequential" for the corresponding class of rational series.

An interesting and relatively well-understood class of weighted automata is obtained by restricting the weights to be taken from a field, or more generally a division ring. This is in fact the historically first setting in which weighted automata were considered, going back to M.-P. Schützenberger [43]. The theory of weighted automata and rational series over fields is characteristic by its use of linear algebra, which becomes a powerful tool – see [8, 41, 42]. In particular, methods of linear algebra underlie a polynomial-time minimisation algorithm for weighted automata over fields due to A. Cardon and M. Crochemore [12]; see also J. Sakarovitch [42] for an exposition. This minimisation algorithm can also be used, e.g., to decide equivalence of rational series over (effective) fields.

Nevertheless, relatively little is known about *deterministic* weighted automata and the sequentiality problem for rational series over fields and their subrings. S. Lombardy and J. Sakarovitch have asked about decidability of whether a rational series over the rational numbers or over the integers, given by a weighted automaton, is sequential [34, Problem 1]. In other words, their question was: Is it decidable whether a given weighted automaton over the rationals or over the integers admits a deterministic equivalent? They have conjectured a positive answer, "at least in the case of one letter alphabet" [34].

There has been little progress on this problem. It has been claimed by S. Lombardy and J. Sakarovitch [34] that decidability of the problem for those *unary* weighted automata over the rationals, whose minimal equivalent automaton consists of *at most two states*, follows by the results on Fibonacci polynomials obtained by G. Jacob, C. Reutenauer, and J. Sakarovitch [26]. In addition, J. Bell and D. Smertnig [7] have recently proved an interesting characterisation of determinisable weighted automata over fields in terms of what they call a *linear hull* of the minimal automaton; however, this characterisation is not effective.

We prove decidability of the determinisability problem for *unary* weighted automata over the rational numbers, the integers, as well as the nonnegative integers – or equivalently, of the sequentiality problem for *univariate* $\mathbb{Q}$-rational, $\mathbb{Z}$-rational, and $\mathbb{N}$-rational series. The former two results confirm the "at least" part of the conjecture of S. Lombardy and J. Sakarovitch [34]. In fact, we observe that the problem can be decided in polynomial time – more precisely, with $O(n^3)$ arithmetic operations over $\mathbb{Q}$ performed, where n is the number of states of the input automaton. We also show that the determinisation of a unary weighted automaton, if possible, can be done algorithmically as well.

To arrive at these results, we first introduce the notion of a *characteristic polynomial* of a unary weighted automaton over the rationals, defined via the characteristic polynomial of the single matrix of its associated linear representation. We show that the minimisation algorithm of A. Cardon and M. Crochemore [12] always computes a reduced representation whose matrix equals the companion matrix of its characteristic polynomial, which we call the characteristic polynomial of the corresponding series. Next, we observe that, in line with similar results from the theory of linear recurrence synthesis [10, Section 7.2], the characteristic polynomial of an automaton is always divided by the characteristic polynomial of its realised series. We then use this property, along with a special form of characteristic polynomials of deterministic automata, to prove a sequentiality criterion for univariate rational series expressed in terms of their characteristic polynomials. An elementary theory of linear difference equations is crucial for obtaining these results.

We finally take a look at the algorithmic side and show that the aforementioned sequentiality criterion is decidable. The decision procedure relies heavily on the theory of cyclotomic polynomials and employs several known algorithms of computer algebra as subroutines.

When it comes to the determinisation algorithm, which is easily implied by the algorithm deciding determinisability, we show that the size of the deterministic automaton it produces is "almost optimal" at least for some specific input automata. The algorithm thus establishes an "almost tight" upper bound for the state complexity of determinisation of unary weighted automata over the rationals.

2. Preliminaries

We denote by $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$, and $\mathbb{C}$, respectively, the sets of nonnegative integers, integers, rational numbers, real numbers, and complex numbers. Given $n \in \mathbb{N}$, we write $[n] = \{1, \ldots, n\}$.

Fields are understood to be commutative, although the theory of weighted automata minimisation, reviewed below, generalises to the noncommutative setting of division rings. Alphabets are assumed to be finite and nonempty. The empty word over any alphabet Σ is denoted by ε. For S a set and $m, n \in \mathbb{N}$, we denote by $S^{m \times n}$ the set of all $m \times n$ matrices over S. The ring of all polynomials in a single variable z, with coefficients from a ring R, is denoted by $R[z]$. The least common multiple of positive integers $p_1, \ldots, p_s$ is denoted by $\operatorname{lcm}(p_1, \ldots, p_s)$ and we write $\gcd(p_1, \ldots, p_s)$ for their greatest common divisor.

2.1. Formal Power Series and Weighted Automata

A *formal power series* in several noncommuting variables from an alphabet Σ with coefficients in a semiring S – or, more briefly, a formal power series over Σ and S – is a mapping $r \colon \Sigma^* \to S$. We write (r, w) instead of $r(w)$ for the value of r upon $w \in \Sigma^*$ and call this element of S the *coefficient* of r at w. The series r itself is then written as

$$r = \sum_{w \in \Sigma^*} (r, w)\, w.$$

The set of all formal power series over Σ and S is denoted by $S\langle\langle \Sigma^* \rangle\rangle$.

The *sum* of power series $r, s \in S\langle\langle \Sigma^* \rangle\rangle$ is the series $r + s$ such that $(r + s, w) = (r, w) + (s, w)$ for all $w \in \Sigma^*$. The reason for abandoning the terminology and notation of mappings, and using those usual for series instead, is related to the multiplicative operation used: The *Cauchy product* of series $r, s \in S\langle\langle \Sigma^* \rangle\rangle$ is the series $r \cdot s$ such that for each $w \in \Sigma^*$,

$$(r \cdot s, w) = \sum_{\substack{u, v \in \Sigma^* \\ uv = w}} (r, u)(s, v).$$

We indentify each $a \in S$ with the series $r_a \in S\langle\langle \Sigma^* \rangle\rangle$ such that $(r_a, \varepsilon) = a$ and $(r, w) = 0$ for all $w \in \Sigma^+$, and each $w \in \Sigma^*$ with the series $r_w \in S\langle\langle \Sigma^* \rangle\rangle$ such that $(r_w, w) = 1$ and $(r_w, x) = 0$ for all $x \in \Sigma^* \setminus \{w\}$. It is well known [17] that $(S\langle\langle \Sigma^* \rangle\rangle, +, \cdot, 0, 1)$ is again a semiring for every semiring S and alphabet Σ.

A family $(r_i \mid i \in I)$ of series from $S\langle\langle \Sigma^* \rangle\rangle$, for an arbitrary index set I, is said to be *locally finite* if the set $I(w) = \{i \in I \mid (r_i, w) \neq 0\}$ is finite for all $w \in \Sigma^*$. One can then define the *sum* over this family by

$$\sum_{i \in I} r_i = r,$$

where $r \in S\langle\langle \Sigma^* \rangle\rangle$ is a series such that the coefficient (r, w) is given, for each $w \in \Sigma^*$, by a *finite* sum

$$(r, w) = \sum_{i \in I(w)} (r_i, w).$$

The *left quotient* of a formal power series $r \in S\langle\langle \Sigma^* \rangle\rangle$ by a word $x \in \Sigma^*$ is a formal power series $x^{-1}r$ such that $(x^{-1}r, w) = (r, xw)$ for all $w \in \Sigma^*$.

Here we are mostly interested in the case when the alphabet Σ is unary, e.g., $\Sigma = \{c\}$, and the coefficients are taken from some field $\mathbb{F}$ or its subring. We obtain the usual univariate formal power series in this case, and the semiring $\mathbb{F}\langle\langle c^* \rangle\rangle$ becomes an integral domain customarily denoted by $\mathbb{F}[[c]]$.

A *weighted finite automaton* over a semiring S and alphabet Σ is a quadruple $\mathcal{A} = (Q, \sigma, \iota, \tau)$, where Q is a finite set of states, $\sigma \colon Q \times \Sigma \times Q \to S$ is a transition weighting function, $\iota \colon Q \to S$ is an initial weighting function, and $\tau \colon Q \to S$ is a terminal weighting function. A *transition* of the automaton $\mathcal{A}$ is a triple $(p, c, q) \in Q \times \Sigma \times Q$ such that $\sigma(p, c, q) \neq 0$. A *run* of the automaton $\mathcal{A}$ is a word $\gamma \in (Q\Sigma)^* Q$ such that (p, c, q) is a transition for all factors pcq of γ such that $p, q \in Q$ and $c \in \Sigma$. Given a run $\gamma = q_0 c_1 q_1 c_2 q_2 \ldots q_{n-1} c_n q_n$ with $n \in \mathbb{N}$, $q_0, \ldots, q_n \in Q$, and $c_1, \ldots, c_n \in \Sigma$, let $\lambda(\gamma) = c_1 c_2 \ldots c_n \in \Sigma^*$ denote the *label* of γ and $\sigma(\gamma) = \sigma(q_0, c_1, q_1)\sigma(q_1, c_2, q_2) \ldots \sigma(q_{n-1}, c_n, q_n) \in S$ the *value* of γ; we also say that γ is a run *from* q_0 *to* q_n. The *monomial* $\|\gamma\| \in S\langle\langle \Sigma^* \rangle\rangle$ realised by the run γ can then be defined by

$$\|\gamma\| = (\iota(q_0)\sigma(\gamma)\tau(q_n))\, \lambda(\gamma).$$

Let $\mathcal{R}(\mathcal{A})$ be the set of all runs of the automaton $\mathcal{A}$. Then it is clear that the family of monomials $(\|\gamma\| \mid \gamma \in \mathcal{R}(\mathcal{A}))$ is locally finite. The *behaviour* of $\mathcal{A}$ can thus be defined by the infinite sum

$$\|\mathcal{A}\| = \sum_{\gamma \in \mathcal{R}(\mathcal{A})} \|\gamma\|.$$

In particular, observe that $\|\mathcal{A}\| = 0$ in case the set Q is empty. A series $r \in S\langle\langle\Sigma^*\rangle\rangle$ is *rational over* S if $r = \|\mathcal{A}\|$ for some weighted finite automaton $\mathcal{A}$ over S and Σ. We often only write that $r \in S\langle\langle\Sigma^*\rangle\rangle$ is *rational*, meaning that r is rational *over* S.

By a *weighted automaton*, we always understand a weighted *finite* automaton. In what follows, we confine ourselves to state sets of the form $Q = [n]$ for some $n \in \mathbb{N}$; this is clearly without loss of generality. Moreover, we write $\mathcal{A} = (n, \sigma, \iota, \tau)$ instead of $\mathcal{A} = ([n], \sigma, \iota, \tau)$.

A weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ is *deterministic* if there is at most one state $q \in Q$ such that $\iota(q) \neq 0$ and if $\sigma(p, c, q) \neq 0$ together with $\sigma(p, c, q') \neq 0$ implies $q = q'$ for all $p, q, q' \in Q$ and $c \in \Sigma$. A series $r \in S\langle\langle\Sigma^*\rangle\rangle$ is *sequential* if $r = \|\mathcal{A}\|$ for some deterministic weighted automaton $\mathcal{A}$ over S and Σ.

Weighted automata *over words* admit an alternative interpretation as *linear representations*. Let S be a semiring and Σ an alphabet. A *linear S-representation* over Σ is a quadruple $\mathcal{P} = (n, \mathbf{i}, \mu, \mathbf{f})$, where $n \in \mathbb{N}$ is its *order*, $\mathbf{i} \in S^{1\times n}$ is a vector of initial weights, $\mu\colon (\Sigma^*, \cdot) \to (S^{n\times n}, \cdot)$ is a monoid homomorphism, and $\mathbf{f} \in S^{n\times 1}$ is a vector of terminal weights. The *series realised by* $\mathcal{P}$ is then defined by

$$\|\mathcal{P}\| = \sum_{w \in \Sigma^*} (\mathbf{i}\mu(w)\mathbf{f})\, w.$$

A series $r \in S\langle\langle\Sigma^*\rangle\rangle$ is *recognisable* if $r = \|\mathcal{P}\|$ for some linear S-representation $\mathcal{P}$ over Σ.

It is a fundamental result that the sets of recognisable and rational series in $S\langle\langle\Sigma^*\rangle\rangle$ coincide [41]. In fact, there is a natural correspondence between weighted automata and linear representations: Given a weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over S and Σ, let $\mathcal{P}_{\mathcal{A}} = (n, \mathbf{i}, \mu, \mathbf{f})$, where $\mathbf{i} = (\iota(1), \ldots, \iota(n))$, $\mu(c) = (a_{i,j})_{n\times n}$ is given for each $c \in \Sigma$ by $a_{i,j} = \sigma(i, c, j)$ for $i, j = 1, \ldots, n$, and $\mathbf{f} = (\tau(1), \ldots, \tau(n))^T$. It is clear that this correspondence introduces a bijection between weighted automata[3] and linear representations over S and Σ, while $\|\mathcal{P}_{\mathcal{A}}\| = \|\mathcal{A}\|$ for every $\mathcal{A}$.

2.2. Minimisation of Weighted Automata over Fields

We now review the basic theory of weighted automata over fields, leading to the Cardon-Crochemore minimisation algorithm [12]. The presentation of this subsection more or less follows J. Sakarovitch [42], where the omitted proofs can be found; see also [8, 30, 41]. References to [42, 8, 41], which would otherwise have been omnipresent, are mostly avoided in what follows.

Let a field $\mathbb{F}$ and an alphabet Σ be fixed for the rest of this subsection. The set $\mathbb{F}\langle\langle\Sigma^*\rangle\rangle$ forms, together with the operations of sum of series and multiplication of a series by a scalar[4] from $\mathbb{F}$, a vector space over $\mathbb{F}$. Given a series $r \in \mathbb{F}\langle\langle\Sigma^*\rangle\rangle$, let the (left) *quotient space* of r be the subspace $\mathcal{Q}(r)$ of $\mathbb{F}\langle\langle\Sigma^*\rangle\rangle$ generated by the set $\{x^{-1}r \mid x \in \Sigma^*\}$ of all left quotients of r. This space is related to minimisation of weighted automata by the following classical result.

Theorem 2.1. *A series $r \in \mathbb{F}\langle\langle\Sigma^*\rangle\rangle$ is rational if and only if the vector space $\mathcal{Q}(r)$ is finite-dimensional. If so, the dimension of $\mathcal{Q}(r)$ equals the minimum number of states of a weighted automaton $\mathcal{A}$ over $\mathbb{F}$ and Σ such that $\|\mathcal{A}\| = r$.*

A weighted automaton $\mathcal{A}$ whose number of states equals the dimension of $\mathcal{Q}(\|\mathcal{A}\|)$ is thus called *minimal*, while the corresponding linear representation is usually termed *reduced*. Note that minimal automata (or, equivalently, reduced representations) are not unique in general.

[3] In case we confine ourselves to state sets of the form $[n]$ for some nonnegative integer n.

[4] Although we have not introduced this operation explicitly, we have identified every scalar a with the power series r_a. Multiplication of $r \in \mathbb{F}\langle\langle\Sigma^*\rangle\rangle$ by $a \in \mathbb{F}$ then coincides with the Cauchy product $r_a \cdot r$.

Two other vector spaces can be associated with each weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ or with the corresponding linear representation $\mathcal{P}_\mathcal{A} = (n, \mathbf{i}, \mu, \mathbf{f})$. The *left vector space* of $\mathcal{A}$ is the subspace $\mathrm{Left}(\mathcal{A})$ of $\mathbb{F}^{1\times n}$ generated by the set $\{\mathbf{i}\mu(w) \mid w \in \Sigma^*\}$. Similarly, the *right vector space* of $\mathcal{A}$ is the subspace $\mathrm{Right}(\mathcal{A})$ of $\mathbb{F}^{n\times 1}$ generated by the set $\{\mu(w)\mathbf{f} \mid w \in \Sigma^*\}$. Moreover, let $\Lambda[\mathcal{A}]\colon \mathrm{Left}(\mathcal{A}) \to \mathbb{F}\langle\!\langle\Sigma^*\rangle\!\rangle$ be a linear mapping given for all $x \in \Sigma^*$ by

$$\Lambda[\mathcal{A}](\mathbf{i}\mu(x)) = \sum_{w\in\Sigma^*} (\mathbf{i}\mu(x)\mu(w)\mathbf{f})\, w.$$

The following theorem then gives two conditions equivalent to the minimality of a weighted automaton.

Theorem 2.2. *Let $\mathcal{A} = (n, \sigma, \iota, \tau)$ be a weighted automaton over $\mathbb{F}$ and Σ. Then the following are equivalent:*

(*i*) *The automaton $\mathcal{A}$ is minimal,* i.e.*, the dimension of $\mathcal{Q}(\|\mathcal{A}\|)$ is n.*

(*ii*) *The dimension of* both *vector spaces* $\mathrm{Left}(\mathcal{A})$ *and* $\mathrm{Right}(\mathcal{A})$ *is n.*

(*iii*) *The dimension of* $\mathrm{Left}(\mathcal{A})$ *is n and $\Lambda[\mathcal{A}]$ is injective.*

Moreover, $\Lambda[\mathcal{A}]$ is injective whenever $\mathrm{Right}(\mathcal{A})$ *is of dimension n.*

One needs a bit more in order to actually *decide* if a given weighted automaton $\mathcal{A}$ over $\mathbb{F}$ and Σ is minimal or not – namely an effective method for computing the dimensions of $\mathrm{Left}(\mathcal{A})$ and $\mathrm{Right}(\mathcal{A})$. To this end, let $\mathcal{P}_\mathcal{A} = (n, \mathbf{i}, \mu, \mathbf{f})$ and call $L \subseteq \Sigma^*$ a *left subbasic language* of $\mathcal{A}$ if it is finite, prefix-closed, and if the family of vectors $(\mathbf{i}\mu(w) \mid w \in L)$ is linearly independent in $\mathrm{Left}(\mathcal{A})$. If in addition $(\mathbf{i}\mu(w) \mid w \in L)$ forms a basis of $\mathrm{Left}(\mathcal{A})$, then call $L \subseteq \Sigma^*$ a *left basic language* of $\mathcal{A}$. *Right subbasic and basic languages* are defined similarly – just replace prefix-closed by suffix-closed, and consider the family $(\mu(w)\mathbf{f} \mid w \in L)$ of vectors from $\mathrm{Right}(\mathcal{A})$ instead.

The most important feature of basic languages is that they actually always exist. In addition, every left (right) subbasic language of $\mathcal{A}$ is a subset of some left (right) basic language of $\mathcal{A}$. This gives rise to a simple iterative algorithm for computing the basic languages and hence also the dimensions of $\mathrm{Left}(\mathcal{A})$ and $\mathrm{Right}(\mathcal{A})$.[5]

The Cardon-Crochemore minimisation algorithm is based on first transforming a weighted automaton $\mathcal{A}$ to its equivalent $\mathcal{B}$ whose number of states equals the dimension of $\mathrm{Right}(\mathcal{B})$. By Theorem 2.2, this also implies that $\Lambda[\mathcal{B}]$ is injective. Next, $\mathcal{B}$ is transformed to its equivalent $\mathcal{C}$ whose number of states equals the dimension $\mathrm{Left}(\mathcal{C})$, while this transformation is conducted in a way that preserves injectivity of $\Lambda[\cdot]$ – that is, $\Lambda[\mathcal{C}]$ is injective as well. The resulting automaton $\mathcal{C}$ is thus minimal by the condition (*iii*) of Theorem 2.2. The details of the construction transforming $\mathcal{B}$ into $\mathcal{C}$ are summarised in the following lemma; the essentially symmetrical transformation of $\mathcal{A}$ into $\mathcal{B}$ is not crucial for our purposes.

Lemma 2.3. *Let $\mathcal{B}$ be a weighted automaton over $\mathbb{F}$ and Σ with $\mathcal{P}_\mathcal{B} = (k, \mathbf{i}, \mu, \mathbf{f})$. Let m be the dimension of* $\mathrm{Left}(\mathcal{B})$ *and assume that $m \geq 1$;*[6] *let $L = \{w_1, \ldots, w_m\}$ with $w_1 = \varepsilon$ be a left basic language of $\mathcal{B}$. Let X be an $m \times k$ matrix with rows $\mathbf{i}\mu(w_1), \ldots, \mathbf{i}\mu(w_m)$, which is obviously of full row rank. Let X_r^{-1} be a right inverse matrix of X.*[7] *Set*

$$\mathbf{i}' = (1, 0, \ldots, 0), \qquad \mu'(c) = X\mu(c)X_r^{-1} \text{ for all } c \in \Sigma, \qquad \text{and} \qquad \mathbf{f}' = X\mathbf{f}.$$

Let $\mathcal{C}$ be the weighted automaton over $\mathbb{F}$ and Σ with $\mathcal{P}_\mathcal{C} = (m, \mathbf{i}', \mu', \mathbf{f}')$. Then $\|\mathcal{C}\| = \|\mathcal{B}\|$ and the dimension of $\mathrm{Left}(\mathcal{C})$ *equals m. Moreover, $\Lambda[\mathcal{C}]$ is injective whenever $\Lambda[\mathcal{B}]$ is.*

[5] The dimension of $\mathrm{Left}(\mathcal{A})$ is clearly equal to the cardinality of any left basic language of $\mathcal{A}$, and similarly for right basic languages.

[6] The case $m = 0$ is trivial and handled separately by the minimisation algorithm.

[7] Note that right inverses might not be unique. However, any of them can be used for our purposes. As the construction of this lemma is later being used in the Cardon-Crochemore algorithm, we assume some deterministic method for computing the right inverses being fixed.

It is in fact possible to prove that the automaton $\mathcal{C}$ is *conjugate* to $\mathcal{B}$ by the matrix X – that is, still using the notation from the previous lemma,

$$\mathbf{i}'X = \mathbf{i}, \qquad \mu'(c)X = X\mu(c) \text{ for all } c \in \Sigma, \qquad \text{and} \qquad \mathbf{f}' = X\mathbf{f}.$$

See also [5, 6] for the theory of weighted automata conjugacy.

The *Cardon-Crochemore minimisation algorithm* [12], which takes upon input a weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over $\mathbb{F}$ and Σ and outputs its minimal equivalent $\mathcal{C}$, consists roughly of the following steps:

1. Find a right basic language R of $\mathcal{A}$. If $R = \emptyset$, output the automaton with no states and halt.
2. Using a construction similar to that of Lemma 2.3, find an automaton $\mathcal{B}$ with k states such that $\|\mathcal{B}\| = \|\mathcal{A}\|$ and $\mathrm{Right}(\mathcal{B})$ is of dimension k.
3. Find a left basic language L of $\mathcal{B}$. If $L = \emptyset$, output the automaton with no states and halt.
4. Using the construction from Lemma 2.3, find and output an automaton $\mathcal{C}$ with m states such that $\|\mathcal{C}\| = \|\mathcal{B}\|$ and $\mathrm{Left}(\mathcal{C})$ is of dimension m.

It can be shown that the algorithm always runs in polynomial time $O(|\Sigma|n^3)$, measured in the number of scalar operations of the field $\mathbb{F}$ performed.

Let $\mathcal{A}, \mathcal{A}'$ be weighted automata over $\mathbb{F}$ and Σ with $\mathcal{P}_{\mathcal{A}} = (n, \mathbf{i}, \mu, \mathbf{f})$ and $\mathcal{P}_{\mathcal{A}'} = (n, \mathbf{i}', \mu', \mathbf{f}')$. These automata are termed *similar* if there exists an invertible matrix $P \in \mathbb{F}^{n\times n}$ such that

$$\mathbf{i} = \mathbf{i}'P, \qquad \mu(c) = P^{-1}\mu'(c)P \text{ for all } c \in \Sigma, \qquad \text{and} \qquad \mathbf{f} = P^{-1}\mathbf{f}'.$$

Similarity thus obviously is a stronger – and symmetric – form of conjugacy. See [41, Proposition III.4.10] for the proof of the following proposition.

Proposition 2.4. *Let $r \in \mathbb{F}\langle\langle\Sigma^*\rangle\rangle$ be a rational series with two* minimal *weighted automata $\mathcal{A}, \mathcal{B}$ such that $\|\mathcal{A}\| = \|\mathcal{B}\| = r$. Then the automata $\mathcal{A}$ and $\mathcal{B}$ are similar.*

The construction of Lemma 2.3 is applied as a final step of the Cardon-Crochemore algorithm unless the realised series is 0. Then $\mathbf{i}'\mu'(w)$ represents, for all $w \in \Sigma^*$, the coordinates of $\mathbf{i}\mu(w)$ relative to the basis $(\mathbf{i}\mu(w_1), \ldots, \mathbf{i}\mu(w_m))$ – indeed, utilising conjugacy of the automata $\mathcal{B}$ and $\mathcal{C}$, it is not hard to prove by induction on the length of w that $\mathbf{i}'\mu'(w)X = \mathbf{i}\mu(w)$. The following proposition, which we state explicitly for later reference, then follows as an easy corollary.

Proposition 2.5. *Let $\mathcal{A}$ be a weighted automaton over $\mathbb{F}$ and Σ, and $\mathcal{B}, \mathcal{C}$ the equivalent automata obtained from $\mathcal{A}$ by the Cardon-Crochemore algorithm, the minimal automaton $\mathcal{C}$ having at least one state. Let $\mathcal{P}_{\mathcal{C}} = (m, \mathbf{i}', \mu', \mathbf{f}')$ and $L = \{w_1, \ldots, w_m\}$ be the left basic language of $\mathcal{B}$ with $w_1 = \varepsilon$. Then $\mathbf{i}'\mu'(w_j) = \mathbf{e}_j$, the j-th vector of the standard basis of $\mathbb{R}^m$, for $j = 1, \ldots, m$. In particular, $\mathbf{i}' = \mathbf{i}'\mu(w_1) = (1, 0, \ldots, 0)$.*[8]

Remark 2.6. Over a unary alphabet $\Sigma = \{c\}$, nonempty left basic languages of automata are necessarily of the form $L = \{\varepsilon, c, c^2, c^3, \ldots, c^m\}$ for some $m \in \mathbb{N}$. For the rest of this article, we assume without loss of generality that the words from L are always numbered in increasing order (with respect to their length) while running the Cardon-Crochemore algorithm: $w_1 = \varepsilon, w_2 = c, \ldots, w_m = c^m$. The equality $\mathbf{i}'\mu'(w_j) = \mathbf{e}_j$ appearing in the preceding proposition thus can be rewritten as $\mathbf{i}'\mu'(c^j) = \mathbf{e}_j$ in this particular case.

2.3. Systems of Difference Equations

We now present some basic facts about linear systems of *difference equations* – *i.e.*, recurrences – that we make use of in what follows. See, e.g., [22, 29] for more information about this topic.

More precisely, we are interested in autonomous and homogeneous systems, which take the form

$$\mathbf{x}_{t+1} = A\mathbf{x}_t \qquad \text{for all } t \in \mathbb{N}, \tag{1}$$

[8] The latter assertion can, of course, be directly seen from the statement of Lemma 2.3 as well.

where $A \in \mathbb{C}^{n\times n}$ is a matrix and $(\mathbf{x}_t)_{t=0}^{\infty}$ with $\mathbf{x}_t = (x_1(t), \ldots, x_n(t))^T$ for all $t \in \mathbb{N}$ is the unknown sequence of vectors. In an *initial value problem* for (1), we are in addition given the vector $\mathbf{x}_0$ and the task is to determine the vectors $\mathbf{x}_t$ for all $t \in \mathbb{N} \setminus \{0\}$. It is easy to see that the unique solution to such initial value problem is always given by $\mathbf{x}_t = A^t\mathbf{x}_0$ for all $t \in \mathbb{N}$. By transforming the matrix A into the Jordan normal form, it is not hard to observe that each $x_j(t)$ for $j \in [n]$ can be expressed as

$$x_j(t) = \sum_{\lambda\in\sigma} \sum_{k=0}^{\alpha(\lambda)-1} b_{\lambda,k} \binom{t}{k} \lambda^{t-k} \qquad \text{for all } t \in \mathbb{N}, \tag{2}$$

where σ denotes the spectrum of A, the algebraic multiplicity of an eigenvalue λ of A is denoted by $\alpha(\lambda)$, and $b_{\lambda,k}$ are complex constants for $\lambda \in \sigma$ and $k = 0, \ldots, \alpha(\lambda) - 1$. Equivalently,

$$x_j(t) = \sum_{\lambda\in\sigma\setminus\{0\}} \sum_{k=0}^{\alpha(\lambda)-1} a_{\lambda,k} t^k \lambda^t + \sum_{\lambda\in\sigma\cap\{0\}} \sum_{k=0}^{\alpha(\lambda)-1} a_{\lambda,k}\delta_{t,k} \qquad \text{for all } t \in \mathbb{N}, \tag{3}$$

where $a_{\lambda,k} \in \mathbb{C}$ for $\lambda \in \sigma$ and $k = 0, \ldots, \alpha(\lambda) - 1$ are constants and $\delta_{a,b}$ is the *Kronecker delta*,

$$\delta_{a,b} = \begin{cases} 1 & \text{if } a = b, \\ 0 & \text{if } a \neq b. \end{cases}$$

Note that the spectrum of A consists precisely of the roots of the *characteristic polynomial* of the matrix A, defined by $\mathrm{ch}_A(z) = \det(z\mathbf{I}_n - A)$, where $\mathbf{I}_n$ is the identity $n\times n$ matrix. In addition, the algebraic multiplicity of an eigenvalue λ is precisely the multiplicity of λ as a root of $\mathrm{ch}_A(z)$. Thus (3) could be rewritten using just roots of $\mathrm{ch}_A(z)$ and their multiplicities – we mostly use this viewpoint in this article.

Now, both (2) and (3) express the function $x_j : \mathbb{N} \to \mathbb{C}$ as a linear combination of other functions from the vector space $\mathbb{C}^{\mathbb{N}}$. It is a fact of fundamental importance that the functions taking place in these linear combinations are linearly independent. More precisely, every finite set of distinct functions of the form

$$f(t) = \binom{t}{k} \lambda^{t-k} \tag{4}$$

for some $\lambda \in \mathbb{C}$ and $k \in \mathbb{N}$ is linearly independent in $\mathbb{C}^{\mathbb{N}}$, as well as every finite set of distinct functions of the form

$$f(t) = t^k\lambda^t \qquad \text{or} \qquad f(t) = \delta_{t,k} \tag{5}$$

for some $\lambda \in \mathbb{C} \setminus \{0\}$ and $k \in \mathbb{N}$. As we need a slightly stronger property than mere linear independence, we briefly sketch the reasoning leading to these observations.

In order to establish linear independence of functions $f_1, \ldots, f_n \in \mathbb{C}^{\mathbb{N}}$, one can utilise the so-called *Casorati matrices*.[9] For each $t \in \mathbb{N}$, define the *Casorati matrix* $\mathrm{Cas}(t)$ of the functions $f_1, \ldots, f_n$ by

$$\mathrm{Cas}(t) = \begin{pmatrix} f_1(t) & f_2(t) & \cdots & f_n(t) \\ f_1(t+1) & f_2(t+1) & \cdots & f_n(t+1) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(t+n-1) & f_2(t+n-1) & \cdots & f_n(t+n-1) \end{pmatrix}.$$

It is not hard to see that $f_1, \ldots, f_n$ are linearly independent whenever $\mathrm{Cas}(t)$ is of full rank for some $t \in \mathbb{N}$. Now, if $f_1, \ldots, f_n$ are pairwise distinct functions of the form λ^t for $\lambda \in \mathbb{C}$, then $\mathrm{Cas}(0)$ is a *Vandermonde matrix*, which is known to have a nonzero determinant – hence $\mathrm{Cas}(0)$ is of full rank, and the said functions are linearly independent. If $f_1, \ldots, f_n$ are functions of the form (4) for $\lambda \in \sigma \subseteq \mathbb{C}$ and $k = 0, \ldots, \alpha(\lambda) - 1$

[9] The determinant of a Casorati matrix, the so-called *Casoratian*, is a discrete counterpart of the Wrońskian, an important tool in the theory of linear *differential* equations.

for each $\lambda \in \sigma$, then $\mathrm{Cas}(0)$ becomes a *generalised Vandermonde matrix*, which is also known to have a nonzero determinant [27]. These functions are thus linearly independent as well. Finally, if $f_1, \ldots, f_n$ are, for some $\sigma \subseteq \mathbb{C}$, functions of the form $t^k \lambda^t$ for $\lambda \in \sigma \setminus \{0\}$ and $k = 0, \ldots, \alpha(\lambda) - 1$, and of the form $\delta_{t,k}$ for $\lambda \in \sigma \cap \{0\}$ and $k = 0, \ldots, \alpha(\lambda) - 1$, then it is easy to see that their Casorati matrix $\mathrm{Cas}(0)$ can be obtained from a generalised Vandermonde matrix via several elementary column operations. Hence, $\mathrm{Cas}(0)$ is of full rank and $f_1, \ldots, f_n$ are linearly independent. The observation that $\mathrm{Cas}(0)$ is of full rank is, in some sense, even more important for our purposes than the actual linear independence of $f_1, \ldots, f_n$. We state this property as a theorem for future reference.

Theorem 2.7. *Let $\sigma \subseteq \mathbb{C}$ be a finite set and $\alpha(\lambda) \in \mathbb{N} \setminus \{0\}$ for each $\lambda \in \sigma$. Let $f_1, \ldots, f_n \colon \mathbb{N} \to \mathbb{C}$ be precisely the functions $f(t) = t^k \lambda^t$ for $\lambda \in \sigma \setminus \{0\}$ and $k = 0, \ldots, \alpha(\lambda) - 1$ together with $f(t) = \delta_{t,k}$ for $\lambda \in \sigma \cap \{0\}$ and $k = 0, \ldots, \alpha(\lambda) - 1$. Then the Casorati matrix $\mathrm{Cas}(0)$ for $f_1, \ldots, f_n$ is of full rank. As a result, any collection of pairwise distinct functions of the form (5) is linearly independent in $\mathbb{C}^{\mathbb{N}}$.*

Let us also recall that the *companion matrix* of a monic polynomial $p(z) = z^n + a_{n-1} z^{n-1} + \ldots + a_1 z + a_0$ with complex coefficients is defined by

$$C_{p(z)} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{pmatrix}.$$

The characteristic polynomial $\mathrm{ch}_{C_{p(z)}}(z)$ of the matrix $C_{p(z)}$ is the polynomial $p(z)$ itself. Linear systems of difference equations with matrices of this type can be used to model linear n-th order difference equations.

2.4. Cyclotomic Polynomials

Let us finally review the basic theory of *cyclotomic polynomials*, later used to decide determinisability of unary weighted automata over $\mathbb{Q}$. See D. S. Dummit and R. M. Foote [21, Section 13.6] for an exposition and missing proofs.

Definition 2.8. Let $n \in \mathbb{N} \setminus \{0\}$. The n-th *cyclotomic polynomial* $\Phi_n(z)$ is then given by

$$\Phi_n(z) = \prod_{\substack{k=1,\ldots,n \\ \gcd(k,n)=1}} \left(z - e^{2k\pi i/n} \right).$$

In other words, $\Phi_n(z)$ is the monic polynomial whose roots are precisely the primitive complex n-th roots of unity. It is a basic fact that actually $\Phi_n(z) \in \mathbb{Z}[z]$ and that this polynomial is always irreducible over $\mathbb{Q}$. This means that cyclotomic polynomials are precisely the minimal polynomials of the primitive roots of unity over $\mathbb{Q}$. The definition also directly implies that the degree of $\Phi_n(z)$ is given by $\varphi(n)$ for all $n \in \mathbb{N} \setminus \{0\}$, where φ denotes the *Euler's totient function* – that is, $\varphi(n)$ is the number of integers from $[n]$ coprime to n. The following formula, which we present as a theorem for later reference, follows in a straightforward manner from Definition 2.8.

Theorem 2.9. *For all $n \in \mathbb{N} \setminus \{0\}$,*

$$z^n - 1 = \prod_{k \mid n} \Phi_k(z).$$

Hence, every cyclotomic polynomial $\Phi_n(z)$ divides $z^p - 1$ for all p that are multiples of n. Moreover, by utilising the irreducibility of cyclotomic polynomials, we easily arrive at the following observation, again presented as a theorem for later reference.

Theorem 2.10. *Let $f(z) \in \mathbb{Q}[z]$ be a monic polynomial such that all roots of $f(z)$ over $\mathbb{C}$ are complex roots of unity. Then $f(z)$ is a product of cyclotomic polynomials. If in addition the roots of $f(z)$ are all simple, then $f(z)$ is a product of pairwise distinct cyclotomic polynomials.*

3. The Characteristic Polynomial of a Unary Weighted Automaton

We now draw our attention to the main objects of our study – that is, to weighted automata over the *rational numbers* and *unary* alphabets, and to the corresponding rational series from $\mathbb{Q}\langle\langle c^* \rangle\rangle$, *i.e.*, $\mathbb{Q}[[c]]$. Of course, as the set $\mathbb{Q}$ forms a field with the usual addition and multiplication, the findings reviewed in Subsection 2.1 and 2.2 still remain valid in this setting.

Let us start by defining the *characteristic polynomials* of unary weighted automata over $\mathbb{Q}$, which we later use as a key ingredient in our determinisability criteria and corresponding decision algorithms. In what follows, we denote by $\mathbf{I}_n$ the $n \times n$ identity matrix over $\mathbb{Q}$.

Definition 3.1. Let $\mathcal{A} = (n, \sigma, \iota, \tau)$ be a weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$ with $\mathcal{P}_{\mathcal{A}} = (n, \mathbf{i}, \mu, \mathbf{f})$ and $n > 0$. The *characteristic polynomial* $\mathrm{ch}_{\mathcal{A}}(z)$ of $\mathcal{A}$ is the characteristic polynomial of the matrix $\mu(c)$, *i.e.*,

$$\mathrm{ch}_{\mathcal{A}}(z) = \mathrm{ch}_{\mu(c)}(z) = \det\left(z\mathbf{I}_n - \mu(c)\right),$$

where $\mathbf{I}_n$ is the $n \times n$ identity matrix.

Remark 3.2. Note that the characteristic polynomial defined in this way is always monic.

Observe that the characteristic polynomial of an automaton does not depend on its initial and final weights, but just on the single matrix $\mu(c)$ of the associated linear representation, with the characteristic polynomial of which it coincides. Now, if $\mathcal{A}$ and $\mathcal{B}$ are similar weighted automata over $\mathbb{Q}$ and $\Sigma = \{c\}$ with $\mathcal{P}_{\mathcal{A}} = (n, \mathbf{i}, \mu, \mathbf{f})$ and $\mathcal{P}_{\mathcal{B}} = (n, \mathbf{i}', \mu', \mathbf{f}')$, then the matrices $\mu(c)$ and $\mu'(c)$ are clearly similar as well. As it is well known that the characteristic polynomials of similar matrices are always equal, we readily arrive at the following observation.

Proposition 3.3. *Let $\mathcal{A}$ and $\mathcal{B}$ be similar unary weighted automata over $\mathbb{Q}$, both having at least one state. Then $\mathrm{ch}_{\mathcal{A}}(z) = \mathrm{ch}_{\mathcal{B}}(z)$.*

Recall Proposition 2.4, according to which any two minimal automata of some fixed rational series over a field are similar. By combining this with Proposition 3.3, we obtain the following corollary.

Corollary 3.4. *Let $r \in \mathbb{Q}\langle\langle c^* \rangle\rangle \setminus \{0\}$ be a rational series. Then all minimal weighted automata $\mathcal{C}$ with $\|\mathcal{C}\| = r$ share the same characteristic polynomial.*

Characteristic polynomials, defined above for *automata*, thus can serve as characteristics of *rational series* as well. This is made explicit by the following definition, in which the notion of a characteristic polynomial is extended to rational series. Correctness of this definition follows by the corollary above.

Definition 3.5. Let $r \in \mathbb{Q}\langle\langle c^* \rangle\rangle \setminus \{0\}$ be a rational series. The *characteristic polynomial* $\mathrm{ch}_r(z)$ of r is the common characteristic polynomial of all minimal automata $\mathcal{C}$ such that $\|\mathcal{C}\| = r$.

Note that the degree of the characteristic polynomial of a rational series always equals the dimension of the quotient space $\mathcal{Q}(r)$.

To compute the characteristic polynomial of a rational series $r \in \mathbb{Q}\langle\langle c^* \rangle\rangle \setminus \{0\}$, it is clearly sufficient to apply the Cardon-Crochemore algorithm in order to obtain a minimal automaton $\mathcal{C}$ such that $\|\mathcal{C}\| = r$, and to subsequently compute the characteristic polynomial of the single matrix of the associated linear representation $\mathcal{P}_{\mathcal{C}}$. However, we now observe that the second step is in fact not necessary, due to a specific form of the automaton obtained via the Cardon-Crochemore algorithm, from which the characteristic polynomial can be "directly read".

Proposition 3.6. *Let $r \in \mathbb{Q}\langle\langle c^* \rangle\rangle \setminus \{0\}$ be a rational series, $\mathcal{A}$ an arbitrary weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$ satisfying $\|\mathcal{A}\| = r$, and $\mathcal{C}$ the minimal automaton for r obtained by the Cardon-Crochemore algorithm upon input $\mathcal{A}$. Let $\mathcal{P}_{\mathcal{C}} = (n, \mathbf{i}, \mu, \mathbf{f})$. Then $\mu(c)$ is the* companion matrix *of the characteristic polynomial* $\mathrm{ch}_r(z) = z^n + a_{n-1}z^{n-1} + \ldots + a_1 z + a_0$ *of the series r,* i.e.,

$$\mu(c) = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{pmatrix}.$$

Proof. By the observation made in Remark 2.6,

$$\mathbf{i}\mu(c^j) = \mathbf{e}_j = (\underbrace{0, \ldots, 0}_{j-1}, 1, \underbrace{0, \ldots, 0}_{n-j})$$

for $j = 1, \ldots, n$. Thus $\mathbf{i} = \mathbf{e}_1$ and the j-th row of $\mu(c)$ is given, for $j = 1, \ldots, n-1$, by the vector $\mathbf{e}_{j+1}$. As a consequence, the first $n-1$ rows of the matrix $\mu(c)$ are as described in the statement of the proposition, which means that

$$\mu(c) = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{pmatrix}$$

for some $a_0, \ldots, a_{n-1} \in \mathbb{Q}$; the minus signs have been introduced just for convenience. The matrix $\mu(c)$ thus is the companion matrix of its characteristic polynomial $\mathrm{ch}_{\mu(c)}(z) = z^n + a_{n-1}z^{n-1} + \ldots + a_1 z + a_0$. Now, $\mathrm{ch}_{\mu(c)}(z) = \mathrm{ch}_{\mathcal{C}}(z)$ and as $\mathcal{C}$ is minimal, $\mathrm{ch}_{\mathcal{C}}(z) = \mathrm{ch}_r(z)$. The matrix $\mu(c)$ therefore indeed is the companion matrix of the characteristic polynomial of r. □

4. The Fundamental Property of Characteristic Polynomials

The characteristic polynomial of a unary weighted automaton may differ from the characteristic polynomial of its realised series in case the automaton is not minimal. Nevertheless, we now prove the following fundamental property of characteristic polynomials of automata, which we later use to obtain a decidable determinisability criterion: Given a unary weighted automaton $\mathcal{A}$ over $\mathbb{Q}$, the characteristic polynomial $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ of the *series* $\|\mathcal{A}\|$ divides the characteristic polynomial $\mathrm{ch}_{\mathcal{A}}(z)$ of the *automaton* $\mathcal{A}$ (if $\|\mathcal{A}\| \neq 0$). In other words, if λ is a root of $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ of multiplicity $\alpha(\lambda)$, then it also is a root of multiplicity at least $\alpha(\lambda)$ for the polynomial $\mathrm{ch}_{\mathcal{A}}(z)$.

Remark 4.1. The main finding of this section is not fully original, as it can be seen, in its essence, as an alternative formulation of certain results from the theory of linear recurrence synthesis centred around the Berlekamp-Massey algorithm – see, e.g., [10, Section 7.2]. Nevertheless, the viewpoint of this section is more expedient for our purposes.

We need the following two lemmata to arrive at the aforementioned observation.

Lemma 4.2. *Let $\mathcal{A} = (n, \sigma, \iota, \tau)$ be a weighted automaton over $\mathbb{Q}$. Then the minimal weighted automaton over $\mathbb{Q}$ equivalent to $\mathcal{A}$ is the same as the minimal weighted automaton over $\mathbb{C}$ equivalent to $\mathcal{A}$.*

Proof. It is easy to see that the Cardon-Crochemore algorithm, presented with an input automaton $\mathcal{A}$ over a field $\mathbb{F}$, produces the same result even if $\mathcal{A}$ is interpreted as an automaton over some extension $\mathbb{K}$ of $\mathbb{F}$. □

A unary weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over $\mathbb{Q}$ and $\Sigma = \{c\}$ with $\mathcal{P}_{\mathcal{A}} = (n, \mathbf{i}, \mu, \mathbf{f})$ and $n > 0$ can clearly be interpreted as an initial problem for the first-order linear autonomous system of difference equations

$$\mathbf{x}_{t+1} = \mu(c)\mathbf{x}_t \qquad \text{for all } t \in \mathbb{N},$$

where the initial conditions are given by $\mathbf{x}_0 = \mathbf{f}$. The j-th component of the column vector $\mathbf{x}_t$ represents, for all $t \in \mathbb{N}$ and $j = 1, \ldots, n$, the coefficient of c^t in the series realised by an automaton obtained from $\mathcal{A}$ by changing the initial weights so that $\iota(j) = 1$ and $\iota(k) = 0$ for $k \in [n] \setminus \{j\}$. Accordingly, the coefficient of c^t in $\|\mathcal{A}\|$ can be expressed, for all $t \in \mathbb{N}$, as

$$\left(\|\mathcal{A}\|, c^t\right) = \mathbf{i} \cdot \mathbf{x}_t.$$

This implies that if we denote by V the set of all complex roots of $\mathrm{ch}_{\mathcal{A}}(z) = \mathrm{ch}_{\mu(c)}(z)$ – *i.e.*, the spectrum of $\mu(c)$ – and by $\alpha(\lambda)$ the multiplicity of a root $\lambda \in V$ of $\mathrm{ch}_{\mathcal{A}}(z)$ – *i.e.*, the algebraic multiplicity of λ as an eigenvalue of $\mu(c)$ – then there exist uniquely determined $a_{\lambda,k} \in \mathbb{C}$ for $\lambda \in V$ and $k = 0, \ldots, \alpha(\lambda) - 1$ such that

$$\left(\|\mathcal{A}\|, c^t\right) = \sum_{\lambda \in V \setminus \{0\}} \sum_{k=0}^{\alpha(\lambda)-1} a_{\lambda,k} t^k \lambda^t + \sum_{\lambda \in V \cap \{0\}} \sum_{k=0}^{\alpha(\lambda)-1} a_{\lambda,k} \delta_{t,k}$$

for all $t \in \mathbb{N}$. We now prove that the constants $a_{\lambda,\alpha(\lambda)-1}$ are nonzero for all $\lambda \in V$ in case the automaton $\mathcal{A}$ is minimal.

Lemma 4.3. *Let $\mathcal{A}$ be a minimal unary weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$ with $\mathcal{P}_{\mathcal{A}} = (n, \mathbf{i}, \mu, \mathbf{f})$ and $n > 0$. Let V be the set of all complex roots of $\mathrm{ch}_{\mathcal{A}}(z)$ and $\alpha(\lambda)$, for each $\lambda \in V$, its multiplicity as a root of $\mathrm{ch}_{\mathcal{A}}(z)$. Then there exist uniquely determined $a_{\lambda,k} \in \mathbb{C}$ for $\lambda \in V$ and $k = 0, \ldots, \alpha(\lambda) - 1$ such that $a_{\lambda,\alpha(\lambda)-1} \neq 0$ for every $\lambda \in V$ and*

$$\left(\|\mathcal{A}\|, c^t\right) = \sum_{\lambda \in V \setminus \{0\}} \sum_{k=0}^{\alpha(\lambda)-1} a_{\lambda,k} t^k \lambda^t + \sum_{\lambda \in V \cap \{0\}} \sum_{k=0}^{\alpha(\lambda)-1} a_{\lambda,k} \delta_{t,k} \qquad \textit{for all } t \in \mathbb{N}. \tag{6}$$

Proof. Existence of *some* uniquely determined constants $a_{\lambda,k} \in \mathbb{C}$ for $\lambda \in V$ and $k = 0, \ldots, \alpha(\lambda) - 1$ satisfying (6) has already been observed above. It remains to prove that $a_{\lambda,\alpha(\lambda)-1}$ is nonzero for each $\lambda \in V$.

Suppose for the purpose of contradiction that $a_{\lambda,\alpha(\lambda)-1} = 0$ for some $\lambda \in V$ and form the polynomial

$$p(z) = (z - \lambda)^{\alpha(\lambda)-1} \prod_{\kappa \in V \setminus \{\lambda\}} (z - \kappa)^{\alpha(\kappa)}$$

of degree $n-1$ with complex coefficients. Let $C \in \mathbb{C}^{(n-1)\times(n-1)}$ be the companion matrix of $p(z)$ and consider a unary weighted automaton $\mathcal{B}$ over $\mathbb{C}$ and $\Sigma = \{c\}$ such that $\mathcal{P}_{\mathcal{B}} = (n-1, \mathbf{i}', \mu', \mathbf{f}')$ for $\mathbf{i}' = (1, 0, \ldots, 0)$, $\mu'(c) = C$, and $\mathbf{f}' = \left((\|\mathcal{A}\|, c^0), \ldots, (\|\mathcal{A}\|, c^{n-2})\right)^T$. We claim that $\|\mathcal{B}\| = \|\mathcal{A}\|$.

For $\kappa \in V$, let $\beta(\kappa)$ be the multiplicity of κ as a root of the polynomial $p(z)$, *i.e.*, $\beta(\kappa) = \alpha(\kappa)$ for $\kappa \neq \lambda$ and $\beta(\lambda) = \alpha(\lambda) - 1$. Clearly $\mathrm{ch}_{\mathcal{B}}(z) = p(z)$, so that there exist uniquely determined constants $b_{\kappa,k} \in \mathbb{C}$ for $\kappa \in V$ and $k = 0, \ldots, \beta(\kappa) - 1$ such that

$$\left(\|\mathcal{B}\|, c^t\right) = \sum_{\kappa \in V \setminus \{0\}} \sum_{k=0}^{\beta(\kappa)-1} b_{\kappa,k} t^k \kappa^t + \sum_{\kappa \in V \cap \{0\}} \sum_{k=0}^{\beta(\kappa)-1} b_{\kappa,k} \delta_{t,k} \tag{7}$$

for all $t \in \mathbb{N}$. The initial value problem corresponding to $\mathcal{B}$ takes the form

$$\mathbf{x}_{t+1} = \mu'(c)\mathbf{x}_t \qquad \text{for all } t \in \mathbb{N}$$

with $\mathbf{x}_0 = \mathbf{f}'$. As $\mu'(c) = C$ is a companion matrix and $\mathbf{i}' = (1, 0, \ldots, 0)$, the solution $\mathbf{x}_t$ satisfies

$$\mathbf{x}_t = \left((\|\mathcal{B}\|, c^t), (\|\mathcal{B}\|, c^{t+1}), \ldots, (\|\mathcal{B}\|, c^{t+n-2})\right)^T$$

for all $t \in \mathbb{N}$. The initial conditions $\mathbf{x}_0 = \mathbf{f}'$ then imply that $(\|\mathcal{B}\|, c^t) = (\|\mathcal{A}\|, c^t)$ for $t = 0, \ldots, n-2$. Combining this observation with (6), (7), and the assumption that $a_{\lambda,\alpha(\lambda)-1} = 0$, we obtain the following system of linear equations in $n-1$ unknowns $a_{\kappa,k} - b_{\kappa,k}$ for $\kappa \in V$ and $k = 0, \ldots, \beta(\kappa)-1$: For $t = 0, \ldots, n-2$,

$$\sum_{\kappa\in V\setminus\{0\}} \sum_{k=0}^{\beta(\kappa)-1} a_{\kappa,k} t^k \kappa^t + \sum_{\kappa\in V\cap\{0\}} \sum_{k=0}^{\beta(\kappa)-1} a_{\kappa,k}\delta_{t,k} = \sum_{\kappa\in V\setminus\{0\}} \sum_{k=0}^{\beta(\kappa)-1} b_{\kappa,k} t^k \kappa^t + \sum_{\kappa\in V\cap\{0\}} \sum_{k=0}^{\beta(\kappa)-1} b_{\kappa,k}\delta_{t,k},$$

or, equivalently,

$$\sum_{\kappa\in V\setminus\{0\}} \sum_{k=0}^{\beta(\kappa)-1} (a_{\kappa,k} - b_{\kappa,k}) t^k \kappa^t + \sum_{\kappa\in V\cap\{0\}} \sum_{k=0}^{\beta(\kappa)-1} (a_{\kappa,k} - b_{\kappa,k})\delta_{t,k} = 0.$$

The matrix of this system is clearly equal to the Casorati matrix $\mathrm{Cas}(0)$ of the functions $t^k\kappa^t$ for $\kappa \in V \setminus \{0\}$ and $k = 0, \ldots, \beta(\kappa) - 1$ and $\delta_{t,k}$ for $k = 0, \ldots, \beta(0) - 1$ in case $0 \in V$. As this matrix is of full rank by Theorem 2.7, we find out that $a_{\kappa,k} - b_{\kappa,k} = 0$ – *i.e.*, $a_{\kappa,k} = b_{\kappa,k}$ – for all $\kappa \in V$ and $k = 0, \ldots, \beta(\kappa) - 1$. Thus indeed $\|\mathcal{A}\| = \|\mathcal{B}\|$ by (6), (7), and the equality $a_{\lambda,\alpha(\lambda)-1} = 0$.

We have thus constructed a unary weighted automaton $\mathcal{B}$ over $\mathbb{C}$ and $\Sigma = \{c\}$ that is smaller than $\mathcal{A}$, while $\|\mathcal{B}\| = \|\mathcal{A}\|$. However, we have assumed that $\mathcal{A}$ is a minimal weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$ with behaviour $\|\mathcal{A}\|$ – a contradiction with Lemma 4.2. □

We are now in a position to prove the fundamental property of characteristic polynomials that we have already anticipated at the beginning of this section.

Theorem 4.4. *Let $\mathcal{A}$ be a unary weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$ with $\|\mathcal{A}\| \neq 0$. Then if λ is a (complex) root of $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ of multiplicity $\alpha(\lambda)$, it is also a root of $\mathrm{ch}_{\mathcal{A}}(z)$ of multiplicity at least $\alpha(\lambda)$.*

Proof. As $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ is by definition the characteristic polynomial of any of the minimal automata for $\|\mathcal{A}\|$ (which is well-defined as $\|\mathcal{A}\| \neq 0$), it follows by Lemma 4.3 that

$$\big(\|\mathcal{A}\|, c^t\big) = \sum_{\lambda\in V\setminus\{0\}} \sum_{k=0}^{\alpha(\lambda)-1} a_{\lambda,k} t^k \lambda^t + \sum_{\lambda\in V\cap\{0\}} \sum_{k=0}^{\alpha(\lambda)-1} a_{\lambda,k}\delta_{t,k} \tag{8}$$

for all $t \in \mathbb{N}$, where V denotes the set of all complex roots of $\mathrm{ch}_{\|\mathcal{A}\|}(z)$, multiplicity of each root $\lambda \in V$ of $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ is denoted by $\alpha(\lambda)$, and $a_{\lambda,k}$, for $\lambda \in V$ and $k = 0, \ldots, \alpha(\lambda) - 1$, are complex numbers such that $a_{\lambda,\alpha(\lambda)-1} \neq 0$ for all $\lambda \in V$. By the discussion preceding Lemma 4.3, we also have

$$\big(\|\mathcal{A}\|, c^t\big) = \sum_{\lambda\in W\setminus\{0\}} \sum_{k=0}^{\beta(\lambda)-1} b_{\lambda,k} t^k \lambda^t + \sum_{\lambda\in W\cap\{0\}} \sum_{k=0}^{\beta(\lambda)-1} b_{\lambda,k}\delta_{t,k} \tag{9}$$

for all $t \in \mathbb{N}$, where W is the set of all complex roots of $\mathrm{ch}_{\mathcal{A}}(z)$, multiplicity of each $\lambda \in W$ as a root of $\mathrm{ch}_{\mathcal{A}}(z)$ is denoted by $\beta(\lambda)$, and $b_{\lambda,k}$, for $\lambda \in W$ and $k = 0, \ldots, \beta(\lambda) - 1$, are some complex constants.

Now, suppose for contradiction that some $\kappa \in V$ is either not a root of $\mathrm{ch}_{\mathcal{A}}(z)$ – *i.e.*, $\kappa \notin W$ – or it is a root of $\mathrm{ch}_{\mathcal{A}}(z)$ of multiplicity smaller than $\alpha(\kappa)$ – *i.e.*, $\beta(\kappa) < \alpha(\kappa)$. In both cases we observe that the right hand side of (8) contains the term

$$F(t) = \begin{cases} a_{\kappa,\alpha(\kappa)-1} t^{\alpha(\kappa)-1}\kappa^t & \text{if } \kappa \neq 0, \\ a_{\kappa,\alpha(\kappa)-1}\delta_{t,\alpha(\kappa)-1} & \text{if } \kappa = 0 \end{cases}$$

with $a_{\kappa,\alpha(\kappa)-1} \neq 0$, while no nonzero factor of this term takes place on the right hand side of (9). As any finite set of distinct functions of the form $f(t) = t^k\lambda^t$ for $k \in \mathbb{N}$ and $\lambda \in \mathbb{C} \setminus \{0\}$ or $f(t) = \delta_{t,k}$ for $k \in \mathbb{N}$ is linearly independent by Theorem 2.7, the right hand sides of (8) and (9) cannot evaluate to the same value for all $t \in \mathbb{N}$. This contradicts the fact that the left hand side is the same both in (8) and in (9). □

Let us end up our preliminary study of characteristic polynomials of automata by restating the preceding theorem in slightly different words.

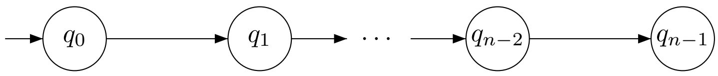
Corollary 4.5. *Let $\mathcal{A}$ be a unary weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$ such that $\|\mathcal{A}\| \neq 0$. Then the polynomial* $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ *divides the polynomial* $\mathrm{ch}_{\mathcal{A}}(z)$.

5. Basic Observations About Deterministic Weighted Automata over Unary Alphabets

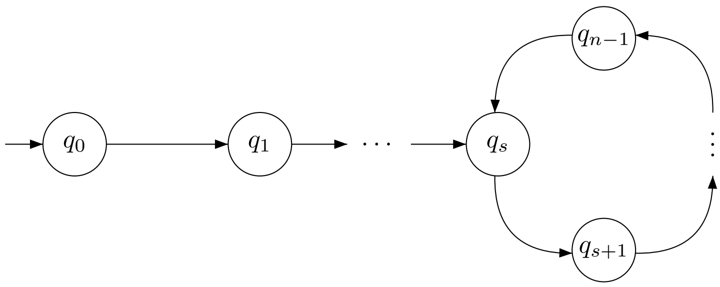
We now take a look at unary *deterministic* weighted automata over $\mathbb{Q}$, which can be assumed to be of a very simple particular form, and identify the characteristic polynomials of such automata.

Call a weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ *accessible* if for every $q \in [n]$, there exists $p \in [n]$ with $\iota(p) \neq 0$ such that there is at least one run γ from p to q.[10] It is clear that every weighted automaton $\mathcal{A}$ admits an accessible equivalent, which is deterministic whenever $\mathcal{A}$ is. As a result, an automaton is determinisable if and only if it admits an *accessible* deterministic equivalent. We may therefore confine ourselves to accessible deterministic automata in what follows.

Let $\mathcal{A} = (n, \sigma, \iota, \tau)$ be an accessible deterministic unary weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$. If $n > 0$, there has to be precisely one state $q_0 \in [n]$ such that $\iota(q_0) \neq 0$. For each $t \in \mathbb{N}$, there clearly exists at most one run γ_t of $\mathcal{A}$ with label c^t from the state q_0. Moreover, the automaton always takes the form of a directed path or a "directed path leading to a cycle"[11] – in the former case, there are distinct states $q_0, \ldots, q_{n-1} \in [n]$ such that γ_t leads to q_t for $t = 0, \ldots, n-1$, while there is no run γ_t for $t \geq n$; in the latter case, there are distinct states $q_0, \ldots, q_{n-1} \in [n]$ and an index $s \in \{0, \ldots, n-1\}$ such that γ_t leads to q_t for $t = 0, \ldots, s-1$ and $\gamma_{t'}$ leads to $q_{s+((t'-s) \bmod (n-s))}$ for all $t' \in \mathbb{N}$ such that $t' \geq s$. The two possibilities are depicted in Fig. 1a and Fig. 1b, respectively.



(a) The first possibility: a directed path.



(b) The second possibility: a "directed path leading to a cycle".

Figure 1: Two possible forms of an accessible deterministic unary weighted automaton over $\mathbb{Q}$ with $n > 0$ states. All transitions are labelled by the same letter c and each is given some nonzero weight. The arrow leading to q_0 indicates that q_0 is the only state with nonzero initial weight; in addition, every state might or might not have a nonzero terminal weight (this is not shown in the figure).

[10] Recall our definition of runs, according to which a run consists of several transitions, each of which has a nonzero weight. Thus necessarily $\sigma(\gamma) \neq 0$ for automata over $\mathbb{Q}$.

[11] The directed path can be of zero length in both cases.

This observation reflects directly in the possible forms of the matrix $\mu(c)$ of the linear representation $\mathcal{P}_\mathcal{A} = (n, \mathbf{i}, \mu, \mathbf{f})$ associated to an accessible deterministic unary weighted automaton $\mathcal{A}$ over $\mathbb{Q}$ and $\Sigma = \{c\}$ with $n > 0$: There always exists a permutation matrix $P \in \{0,1\}^{n\times n}$ such that

$$P\mu(c)P^{-1} = \begin{pmatrix} 0 & a_{1,2} & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & a_{2,3} & \cdots & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a_{s-1,s} & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & a_{s,s+1} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & a_{n-2,n-1} & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & a_{n-1,n} \\ 0 & 0 & 0 & \cdots & a_{n,s} & 0 & \cdots & 0 & 0 \end{pmatrix}, \tag{10}$$

where $s \in [n]$, $a_{j,j+1} \in \mathbb{Q} \setminus \{0\}$ for $j = 1, \ldots, n-1$, and $a_{n,s} \in \mathbb{Q}$, while these elements are in $\mathbb{Z}$ or $\mathbb{N}$ whenever $\mathcal{A}$ is an automaton over $\mathbb{Z}$ or $\mathbb{N}$, respectively.[12]

Characteristic polynomials of matrices taking the form just described can be easily determined – we thus arrive at the following result.

Proposition 5.1. *Let $\mathcal{A} = (n, \sigma, \iota, \tau)$ with $n > 0$ be an accessible deterministic unary weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$. Then either*

$$\mathrm{ch}_\mathcal{A}(z) = z^n,$$

or

$$\mathrm{ch}_\mathcal{A}(z) = z^n - bz^k$$

for some $b \in \mathbb{Q} \setminus \{0\}$ and $k \in \{0, \ldots, n-1\}$. In the latter case, $b \in \mathbb{Z} \setminus \{0\}$ whenever $\mathcal{A}$ is an automaton over $\mathbb{Z}$, and $b \in \mathbb{N} \setminus \{0\}$ whenever $\mathcal{A}$ is an automaton over $\mathbb{N}$.

Proof. Let $\mathcal{P}_\mathcal{A} = (n, \mathbf{i}, \mu, \mathbf{f})$. Then (10) has to hold for some permutation matrix $P \in \{0,1\}^{n\times n}$, $s \in [n]$, $a_{j,j+1} \in \mathbb{Q}\setminus\{0\}$ for $j = 1, \ldots, n-1$, and $a_{n,s} \in \mathbb{Q}$; if $\mathcal{A}$ is over $\mathbb{Z}$ or $\mathbb{N}$, then these elements belong to the same set as well. Clearly

$$\mathrm{ch}_\mathcal{A}(z) = \mathrm{ch}_{P\mu(c)P^{-1}}(z) = z^{s-1}\left(z^{n-s+1} - a_{n,s}\prod_{j=s}^{n-1} a_{j,j+1}\right) = z^n - a_{n,s}z^{s-1}\prod_{j=s}^{n-1} a_{j,j+1}. \tag{11}$$

If $a_{n,s} = 0$, then (11) boils down to $\mathrm{ch}_\mathcal{A}(z) = z^n$. Otherwise, set

$$b = a_{n,s}\prod_{j=s}^{n-1} a_{j,j+1}$$

and $k = s - 1$; the equation (11) then rewrites as $\mathrm{ch}_\mathcal{A}(z) = z^n - bz^k$. □

Corollary 5.2. *Let $\mathcal{A} = (n, \sigma, \iota, \tau)$ with $n > 0$ be an accessible deterministic unary weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$. Then the roots of the characteristic polynomial $\mathrm{ch}_\mathcal{A}(z)$ can be described as follows:*

(*i*) *Zero can be a possibly multiple root of $\mathrm{ch}_\mathcal{A}(z)$.*

(*ii*) *If zero is not a root of $\mathrm{ch}_\mathcal{A}(z)$ of multiplicity n, let $p = n$ if it is not a root of $\mathrm{ch}_\mathcal{A}(z)$ at all, and $p = n-k$ if it is a root of multiplicity k. Then there exists $b \in \mathbb{Q} \setminus \{0\}$ such that the remaining roots of $\mathrm{ch}_\mathcal{A}(z)$ are all simple and given by $\beta \cdot e^{2\ell\pi i/p}$ for $\ell = 0, \ldots, p-1$ and any $\beta \in \mathbb{C}$ such that $\beta^p = b$.*

[12] If $s = 0$, then $a_{n,s}$ is in the first column and there is no element $a_{s-1,s}$.

6. Deciding Determinisability and Determinisation

The aim of this section is to collect our hitherto findings and to proceed to the actual main result of this article – a decidable determinisability criterion for unary weighted automata over $\mathbb{Q}$ – and to some related points, such as the variant of the main result for automata over $\mathbb{Z}$ and $\mathbb{N}$, or the existence of a determinisation algorithm. Nevertheless, we first need one further auxiliary proposition, which can be viewed as a stronger[13] converse of Proposition 5.1 and Corollary 5.2.

Proposition 6.1. *Let $r \in \mathbb{Q}\langle\langle c^*\rangle\rangle \setminus \{0\}$ be a rational series.*

(i) If the characteristic polynomial $\mathrm{ch}_r(z)$ divides z^q for some $q \in \mathbb{N}\setminus\{0\}$, then $r = \|\mathcal{A}\|$ for a deterministic unary weighted automaton $\mathcal{A}$ over $\mathbb{Q}$ and $\Sigma = \{c\}$ with $\mathcal{P}_\mathcal{A} = (q, \mathbf{i}, \mu, \mathbf{f})$, where $\mathbf{i} = (1, 0, \ldots, 0)$, $\mu(c)$ is the companion matrix of z^q, i.e.,

$$\mu(c) = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix},$$

and $\mathbf{f} = \big((r, c^0), (r, c^1), \ldots, (r, c^{q-1})\big)^T$.

(ii) If $\mathrm{ch}_r(z)$ divides $z^q - bz^h$ for some $q \in \mathbb{N} \setminus \{0\}$, $b \in \mathbb{Q} \setminus \{0\}$, and $h \in \{0, \ldots, q-1\}$ – i.e., *the roots of $\mathrm{ch}_r(z)$ are elements of the set $\{\beta \cdot e^{2k\pi i/p} \mid k \in \{0, \ldots, p-1\}\} \cup \{0\}$ for $p = q-h$ and some $\beta \in \mathbb{C}$ such that $\beta^p = b$, while all these roots, possibly except zero, are simple – then $r = \|\mathcal{A}\|$ for a deterministic unary weighted automaton $\mathcal{A}$ over $\mathbb{Q}$ and $\Sigma = \{c\}$ with $\mathcal{P}_\mathcal{A} = (q, \mathbf{i}, \mu, \mathbf{f})$, where $\mathbf{i} = (1, 0, \ldots, 0)$, $\mu(c)$ is the companion matrix of $z^q - bz^h$,* i.e.,

$$\mu(c) = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 1 \\ \underbrace{0 \quad 0 \quad 0 \quad \cdots \quad b}_{h+1} & & & & & 0 & \cdots & 0 & 0 \end{pmatrix},$$

and $\mathbf{f} = \big((r, c^0), (r, c^1), \ldots, (r, c^{q-1})\big)^T$.

Proof. The statement (i) is obviously true; let us thus prove (ii). Let

$$V = \{\beta \cdot e^{2k\pi i/p} \mid k \in \{0, \ldots, p-1\}\} \cup \{0\}.$$

Moreover, let $V' \subseteq V$ be the set of actual roots of $\mathrm{ch}_r(z)$, and $h' \leq h$ the multiplicity of zero as a root of $\mathrm{ch}_r(z)$ (or 0 if zero is not a root of $\mathrm{ch}_r(z)$ at all). By the reasoning preceding Lemma 4.3, we obtain

$$(r, c^t) = \sum_{\lambda \in V' \setminus \{0\}} a_{\lambda,0} \lambda^t + \sum_{k=0}^{h'-1} a_{0,k} \delta_{t,k}$$

[13] It is just assumed in its statement that $\mathrm{ch}_r(z)$ *divides* z^q or $z^q - bz^h$.

for some constants $a_{\lambda,0} \in \mathbb{C}$ for $\lambda \in V' \setminus \{0\}$ and $a_{0,0}, \ldots, a_{0,h'-1} \in \mathbb{C}$, and for all $t \in \mathbb{N}$. By setting $a_{\lambda,0} = 0$ for all $\lambda \in V \setminus V'$ and $a_{0,h'} = a_{0,h'+1} = \ldots = a_{0,h-1} = 0$, this can be rewritten as

$$(r, c^t) = \sum_{\lambda \in V \setminus \{0\}} a_{\lambda,0} \lambda^t + \sum_{k=0}^{h-1} a_{0,k} \delta_{t,k} \tag{12}$$

for all $t \in \mathbb{N}$. Now, as $\mu(c)$ is the companion matrix of $z^q - bz^h$, all elements of $V \setminus \{0\}$ are simple roots of $\mathrm{ch}_{\mathcal{A}}(z)$ and if $h > 0$, zero is a root of multiplicity h. Thus, again by the reasoning preceding Lemma 4.3,

$$(\|\mathcal{A}\|, c^t) = \sum_{\lambda \in V \setminus \{0\}} b_{\lambda,0} \lambda^t + \sum_{k=0}^{h-1} b_{0,k} \delta_{t,k} \tag{13}$$

for some constants $b_{\lambda,0} \in \mathbb{C}$ for $\lambda \in V \setminus \{0\}$ and $b_{0,0}, \ldots, b_{0,h-1} \in \mathbb{C}$, and for all $t \in \mathbb{N}$. Moreover, clearly

$$(\|\mathcal{A}\|, c^t) = (r, c^t) \qquad \text{for } t = 0, \ldots, q-1.$$

Thus, by (12) and (13),

$$\sum_{\lambda \in V \setminus \{0\}} a_{\lambda,0} \lambda^t + \sum_{k=0}^{h-1} a_{0,k} \delta_{t,k} = \sum_{\lambda \in V \setminus \{0\}} b_{\lambda,0} \lambda^t + \sum_{k=0}^{h-1} b_{0,k} \delta_{t,k} \qquad \text{for } t = 0, \ldots, q-1.$$

As the $q \times q$ Casorati matrix $\mathrm{Cas}(0)$ of the functions λ^t for $\lambda \in V \setminus \{0\}$ and $\delta_{t,k}$ for $k = 0, \ldots, h-1$ is of full rank by Theorem 2.7, we obtain $a_{\lambda,0} = b_{\lambda,0}$ for all $\lambda \in V$ and $a_{0,k} = b_{0,k}$ for $k = 0, \ldots, h-1$. Hence, $(\|\mathcal{A}\|, c^t) = (r, c^t)$ for all $t \in \mathbb{N}$ – or, in other words, $\|\mathcal{A}\| = r$. □

We are now prepared to prove the main theorem of this article providing a determinisability criterion for unary weighted automata over the rationals, which we later show to be decidable in polynomial time. To be more precise, we formulate this criterion in three different, yet obviously equivalent, ways, as conditions *(ii)* to *(iv)* of the following theorem. We still assume that the behaviour of the automaton in question is a nonzero series, as otherwise the automaton is trivially determinisable.

Theorem 6.2. *Let $\mathcal{A} = (n, \sigma, \iota, \tau)$ with $\|\mathcal{A}\| \neq 0$ be a unary weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$. Then the following are equivalent:*

(i) *The automaton $\mathcal{A}$ is determinisable,* i.e., *there exists a deterministic weighted automaton $\mathcal{B}$ over $\mathbb{Q}$ and $\Sigma = \{c\}$ such that $\|\mathcal{B}\| = \|\mathcal{A}\|$.*

(ii) *The characteristic polynomial $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ of the* series *realised by the automaton $\mathcal{A}$ divides either z^q for some $q \in \mathbb{N} \setminus \{0\}$, or $z^q - bz^h$ for some $q \in \mathbb{N} \setminus \{0\}$, $b \in \mathbb{Q} \setminus \{0\}$, and $h \in \{0, \ldots, q-1\}$.*

(iii) *The characteristic polynomial $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ can be written either as*

$$\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k$$

for some $k \in \mathbb{N} \setminus \{0\}$, or as

$$\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k f(z),$$

where $k \in \mathbb{N}$ and $f(z) \in \mathbb{Q}[z]$ is a monic polynomial of degree $d > 0$ that divides $z^p - b$ for some $p \in \mathbb{N} \setminus \{0\}$ and $b \in \mathbb{Q} \setminus \{0\}$.

(iv) *The roots of $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ all belong to the set $\{\beta \cdot e^{2k\pi i/p} \mid k \in \{0, \ldots, p-1\}\} \cup \{0\}$ for some $p \in \mathbb{N} \setminus \{0\}$ and $\beta \in \mathbb{C}$ such that $\beta^p \in \mathbb{Q}$, while all these roots, possibly except zero, are simple.*

Proof. The statement (*ii*) is clearly equivalent both to (*iii*) via $z^q - bz^h = z^h(z^{q-h} - b)$ and $p = q - h$, and to (*iv*) via $\beta = 0$ if $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ divides some z^q, eventually via $p = q - h$ and $\beta^p = b$ otherwise.

It thus remains to prove the equivalence of, e.g., (*i*) and (*ii*). In case the automaton $\mathcal{A}$ is determinisable, there exists an *accessible* deterministic weighted automaton $\mathcal{B} = (q, \sigma', \iota', \tau')$ over $\mathbb{Q}$ and $\Sigma = \{c\}$ such that $\|\mathcal{B}\| = \|\mathcal{A}\|$; as $\|\mathcal{A}\| \neq 0$, necessarily $q > 0$. Then either

$$\mathrm{ch}_{\mathcal{B}}(z) = z^q,$$

or

$$\mathrm{ch}_{\mathcal{B}}(z) = z^q - bz^h$$

for some $b \in \mathbb{Q} \setminus \{0\}$ and $h \in \{0, \ldots, q-1\}$ by Proposition 5.1, and $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ divides $\mathrm{ch}_{\mathcal{B}}(z)$ by Corollary 4.5.

Conversely, if $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ divides z^q or $z^q - bz^h$ for some $q \in \mathbb{N} \setminus \{0\}$, $b \in \mathbb{Q} \setminus \{0\}$, and $h \in \{0, \ldots, q-1\}$, then $\mathcal{A}$ is determinisable by Proposition 6.1. □

Our next aim is to show that the equivalent conditions given by the preceding theorem are decidable, *i.e.*, to describe an algorithm deciding determinisability of unary weighted automata over $\mathbb{Q}$.

Let $\mathcal{A} = (n, \sigma, \iota, \tau)$ be a unary weighted automaton over $\mathbb{Q}$ and $\Sigma = \{c\}$, determinisability of which is in question. By applying the Cardon-Crochemore minimisation algorithm, it is easy to decide if $\|\mathcal{A}\| = 0$, as this happens if and only if the equivalent minimal automaton $\mathcal{B} = (m, \sigma', \iota', \tau')$ is empty, *i.e.*, if $m = 0$. If so, the automaton $\mathcal{A}$ is determinisable. In the opposite case the minimisation algorithm computes, in the sense of Proposition 3.6, the characteristic polynomial $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ of the series $\|\mathcal{A}\|$. It is then trivial to decide whether $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k$ for some $k \in \mathbb{N} \setminus \{0\}$, in which case $\mathcal{A}$ is determinisable by Theorem 6.2.[14]

We may thus suppose that the characteristic polynomial $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ can be factored as $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k f(z)$ for some $k \in \mathbb{N}$ and some monic polynomial $f(z) \in \mathbb{Q}[z]$ of degree $d > 0$ with nonzero constant coefficient. Deciding whether $\mathcal{A}$ is determinisable reduces, by Theorem 6.2, to finding out if $f(z)$ divides $z^p - b$ for some $p \in \mathbb{N} \setminus \{0\}$ and $b \in \mathbb{Q} \setminus \{0\}$. As $(z^p - b)(z^p + b) = z^{2p} - b^2$, we may assume without loss of generality that actually $b > 0$. The condition of $f(z)$ dividing $z^p - b$ for some $p \in \mathbb{N} \setminus \{0\}$ and positive $b \in \mathbb{Q}$ is equivalent to saying that the roots of $f(z)$ are all simple and contained in the set $\{\sqrt[p]{b} \cdot e^{2k\pi i/p} \mid k \in \{0, \ldots, p-1\}\}$ for some $p \in \mathbb{N} \setminus \{0\}$ and positive $b \in \mathbb{Q}$, where $\sqrt[p]{b}$ denotes the *positive real* p-th root of b.

Let us now write the polynomial $f(z)$ as

$$f(z) = z^d + a_{d-1}z^{d-1} + \ldots + a_1 z + a_0$$

for $a_0, a_1, \ldots, a_{d-1} \in \mathbb{Q}$ with $a_0 \neq 0$. Then if the roots of $f(z)$ indeed all take the form $\sqrt[p]{b} \cdot e^{2k\pi i/p}$ for some k, we necessarily get $|a_0| = b^{d/p}$, *i.e.*, $\sqrt[p]{b} = \sqrt[d]{|a_0|}$.

Consider the companion matrix C of the polynomial $f(z)$,

$$C = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{d-1} \end{pmatrix},$$

whose eigenvalues are precisely the roots of $f(z)$. If the roots of $f(z)$ are of the form described above, all eigenvalues of the matrix

$$A = \frac{1}{|a_0|} C^d$$

must be (not necessarily distinct) roots of unity – in other words, applying Theorem 2.10, the characteristic polynomial $\mathrm{ch}_A(z) \in \mathbb{Q}[z]$ is a product of (not necessarily distinct) cyclotomic polynomials. Thus, if we find out that this characteristic polynomial is *not* a product of cyclotomic polynomials, we can conclude that the automaton $\mathcal{A}$ is not determinisable.

[14]Obviously $k = m$ in this case.

In case the polynomial $\mathrm{ch}_A(z)$ *is* a product of cyclotomic polynomials, write

$$\mathrm{ch}_A(z) = \prod_{j=1}^{s} \Phi_{p_j}(z),$$

where $s, p_1, \ldots, p_s \in \mathbb{N} \setminus \{0\}$. Then each $\Phi_{p_j}(z)$, for $j = 1, \ldots, s$, gives $\varphi(p_j)$ roots of the polynomial $\mathrm{ch}_A(z)$, which are all of the form $e^{2k\pi i/p_j}$ for some $k \in \{0, \ldots, p_j - 1\}$;[15] denote the set of all these roots by V_j. All roots of $\mathrm{ch}_A(z)$ are thus of the form $e^{2k\pi i/\,\mathrm{lcm}(p_1,\ldots,p_s)}$ for some $k \in \{0, \ldots, \mathrm{lcm}(p_1, \ldots, p_s) - 1\}$.

Each root $\omega = e^{2k\pi i/p_j} \in V_j$ for $j = 1, \ldots, s$ surely corresponds to exactly one root of $f(z)$,[16] which has to be $\sqrt[d]{|a_0|} \cdot e^{2(k+\ell p_j)\pi i/(p_j d)}$ for some $\ell \in \{0, \ldots, d-1\}$, *i.e.*, $\sqrt[d]{|a_0|}$ times one of the d-th roots of ω. This observation can be used to determine all roots of the polynomial $f(z)$, but there is no need to do so. Instead, as we shortly demonstrate, it suffices find out whether all roots of $f(z)$ are simple. This can be easily done via computing the greatest common divisor of $f(z)$ with its formal derivative $f'(z)$ – see, for instance, [21, Section 13.5].

In case there are repeated roots of $f(z)$, we may conclude by Theorem 6.2 that $\mathcal{A}$ is not determinisable. Otherwise the roots of $f(z)$ are all simple, and we may set $p = d \cdot \mathrm{lcm}(p_1, \ldots, p_s)$ and $b = |a_0|^{p/d}$. We then find out that all these roots are contained in the set

$$\{\sqrt[p]{b} \cdot e^{2k\pi i/p} \mid k \in \{0, \ldots, p-1\}\}.$$

Moreover, the number b is necessarily rational, as $b = |a_0|^{p/d} = |a_0|^{d\cdot\mathrm{lcm}(p_1,\ldots,p_s)/d} = |a_0|^{\mathrm{lcm}(p_1,\ldots,p_s)} \in \mathbb{Q}$. We may thus conclude, again by Theorem 6.2, that the automaton $\mathcal{A}$ is determinisable.

Let us now summarise the algorithm for deciding determinisability of unary weighted automata over $\mathbb{Q}$. We first provide a high-level description, while the details concerning implementation of certain particular steps can be found below.

Algorithm 6.3 (Deciding determinisability of unary weighted automata over $\mathbb{Q}$).

Input: A unary weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over $\mathbb{Q}$ and $\Sigma = \{c\}$.

1. Find a minimal automaton $\mathcal{B} = (m, \sigma', \iota', \tau')$ equivalent to $\mathcal{A}$ using the Cardon-Crochemore algorithm.
2. Check if $m = 0$.
   a) If so, return "$\mathcal{A}$ *is determinisable*" and halt.
   b) Otherwise "read" the characteristic polynomial $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ and continue by the next step.
3. Check if $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k$ for some $k \in \mathbb{N} \setminus \{0\}$.
   a) If so, return "$\mathcal{A}$ *is determinisable*" and halt.
   b) Otherwise continue by the next step.
4. Compute the factorisation $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k f(z)$, where $k \in \mathbb{N}$ and $f(z)$ is a monic polynomial of degree $d > 0$ with nonzero constant coefficient a_0.
5. Compute the matrix $A = (1/|a_0|)C^d$, where C is the companion matrix of $f(z)$.
6. Compute the characteristic polynomial $\mathrm{ch}_A(z)$ of A.
7. Check if $\mathrm{ch}_A(z)$ is a product of (not necessarily distinct) cyclotomic polynomials.
   a) If not, return "$\mathcal{A}$ *is not determinisable*" and halt.
   b) Otherwise continue by the next step.
8. Check if all roots of $f(z)$ are simple.
   a) If so, return "$\mathcal{A}$ *is determinisable*".
   b) Otherwise return "$\mathcal{A}$ *is not determinisable*".

[15] Of course, not all values of k are actually possible here in general.

[16] More precisely, to a root λ such that $\lambda^d/|a_0| = \omega$. If λ is a multiple root, then ω is a root of at least the same multiplicity, so that each λ indeed has an ω to which it corresponds. Similarly in the case of multiple λ with the same λ^d.

Some comments are in place as to the implementation of certain steps of the above algorithm and their time complexity, measured by the number of arithmetic operations over $\mathbb{Q}$. Regarding time complexity, our aim is to show that at most $O(n^3)$ such operations are performed during any execution of Algorithm 6.3 upon an input automaton with n states, in case a suitable implementation is used.

Step 1 of Algorithm 6.3 consists of calling the Cardon-Crochemore minimisation algorithm, which runs with $O(n^3)$ arithmetic operations – and, as we show below, it is the least effective step of the algorithm. The way how to "read" the characteristic polynomial $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ in Step 2 from the minimal automaton computed by the Cardon-Crochemore algorithm was explained in Proposition 3.6. The next two steps of the algorithm are trivial and negligible when it comes to time complexity.

The cost of computing the companion matrix C in Step 5 is clearly negligible. The power C^d of the $d \times d$ matrix C can clearly be computed within

$$O(M(d)\log d) = O(M(n)\log n)$$

arithmetic operations, where $M(n)$ is the number of operations needed for multiplication of $n \times n$ matrices. Thus, if the matrix multiplication is realised, e.g., by the Strassen algorithm [44] or by the "fastest" algorithm known up to date [2], the arithmetic complexity of Step 5 goes well below $O(n^3)$. Similarly, the characteristic polynomial of a $d \times d$ matrix can be computed in $O(M(d)\log d)$ operations [28, 20], so we obtain the same bound for Step 6.

Step 7 can be considered the crux of Algorithm 6.3. There is a variety of methods that could be employed in order to check whether a polynomial is or is not a product of cyclotomic polynomials. Let us first describe an elementary approach, for which an $O(n^3)$ upper bound on arithmetic operations is easily established. First observe that since $\mathrm{ch}_A(z)$ is of degree d, every cyclotomic polynomial dividing $\mathrm{ch}_A(z)$ has to be of degree at most d. The degree of a cyclotomic polynomial $\Phi_k(z)$ equals $\varphi(k)$. At the same time, it is known that

$$\varphi(k) > \frac{k}{e^{\gamma}\ln\ln k + 3/(\ln\ln k)}$$

for all $k \geq 3$, where γ is the Euler–Mascheroni constant – see [40, Theorem 15] and [4, Theorem 8.8.7]. This implies existence of an effectively computable function $B(d) = O\left(d^{1+\delta}\right)$ with $0 < \delta < 1/2$ such that $k \leq B(d)$ for all cyclotomic polynomials $\Phi_k(z)$ dividing $\mathrm{ch}_A(z)$ of degree d. We can then perform trial division of $\mathrm{ch}_A(z)$ by the cyclotomic polynomials $\Phi_1(z), \ldots, \Phi_{B(d)}(z)$, in this order, while attempting division by each polynomial possibly multiple times until a nonzero remainder is obtained. At the end of this process, we surely obtain all cyclotomic factors of the polynomial $\mathrm{ch}_A(z)$, so that it finally suffices to check whether there is a factor of positive degree remaining or not. The whole procedure thus involves computing the cyclotomic polynomials $\Phi_1(z), \ldots, \Phi_{B(d)}(z)$, while it is well known that $\Phi_k(z)$ can be computed using $O(k(\log k)^2(\log\log k))$ operations – see, e.g., J. von zur Gathen and J. Gerhard [45, Algorithm 14.48]; for other possibilities of computing the cyclotomic polynomials, consult A. Arnold and M. Monagan [3]. The overall number of arithmetic operations needed to compute $\Phi_1(z), \ldots, \Phi_{B(d)}(z)$ thus is in

$$O\Big(B(d)^2\,(\log B(d))^2\,(\log\log B(d))\Big) = O\Big(d^{2+2\delta}\,(\log d)^2\,(\log\log d)\Big) = O\big(d^3\big) = O\big(n^3\big)\,.$$

The division with remainder can be done with $O(d(\log d)(\log\log d))$ operations [45, Algorithm 9.5][17] for polynomials of degree bounded by d, while the method described performs at most d successful and at most $B(d)$ unsuccessful divisions. In effect, the total number of operations needed to perform the divisions is in

$$O(B(d)d\,(\log d)\,(\log\log d)) = O\big(d^{2+\delta}\,(\log d)\,(\log\log d)\big) = O\big(d^3\big) = O\big(n^3\big)\,.$$

As a result, we may conclude that at most $O(n^3)$ arithmetic operations over $\mathbb{Q}$ are needed for Step 7.

[17] As we are interested in solving the problem using arithmetic operations of $\mathbb{Q}$ only, we do not take into account the methods relying on the FFT.

However, note that there is a possibly faster approach than the one described above: F. Beukers and C. J. Smyth [9] have described an effective algorithm, based upon similar ideas as the "Graeffe" cyclotomicity test of R. J. Bradford and J. H. Davenport [11], for finding what they call the *cyclotomic part* of a given polynomial $g(z)$, *i.e.*, the product of all *distinct* cyclotomic polynomials dividing $g(z)$. Now, it is clear that a polynomial is a product of cyclotomic polynomials if and only if its square-free part is a product of distinct cyclotomic polynomials. Hence, to find out whether $\mathrm{ch}_A(z)$ is a product of cyclotomic polynomials, it suffices to compute its square-free part – *i.e.*, to divide $\mathrm{ch}_A(z)$ by the greatest common divisor of $\mathrm{ch}_A(z)$ and its formal derivative [45, Algorithm 14.19] – and to apply the aforementioned algorithm of F. Beukers and C. J. Smyth [9].

Finally, Step 8 can be realised via computing the greatest common divisor of $f(z)$ and its formal derivative – the roots of $f(z)$ are all simple if and only if the result is of degree zero [21, Section 13.5]. This can be done within

$$O\Big(d\,(\log d)^2\,(\log\log d)\Big) = O\big(d^3\big) = O\big(n^3\big)$$

arithmetic operations over the rational numbers [45, Algorithm 14.19]. We may thus conclude as follows.

Theorem 6.4. *Given any input automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$, the Algorithm 6.3 correctly decides whether $\mathcal{A}$ is determinisable, while $O(n^3)$ arithmetical operations over $\mathbb{Q}$ are performed.*

Let us now collect some closely related observations. In the first place, let us note that determinisability of unary weighted automata over $\mathbb{Z}$ and $\mathbb{N}$ is decidable as well. We need the following variant of Theorem 6.2 in order to arrive at this observation.

Proposition 6.5. *Let $\mathcal{A} = (n, \sigma, \iota, \tau)$ with $\|\mathcal{A}\| \neq 0$ be a unary weighted automaton over $\mathbb{Z}$ and $\Sigma = \{c\}$. Then the following are equivalent:*

(*i*) *The automaton $\mathcal{A}$ is determinisable over $\mathbb{Z}$.*

(*ii*) *The characteristic polynomial $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ of $\|\mathcal{A}\|$ satisfies either $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k$ for some $k \in \mathbb{N} \setminus \{0\}$, or $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k f(z)$ for $k \in \mathbb{N}$ and a monic polynomial $f(z) \in \mathbb{Q}[z]$ of degree $d > 0$ dividing $z^p - b$ for some $p \in \mathbb{N} \setminus \{0\}$ and $b \in \mathbb{Z} \setminus \{0\}$.*

(*iii*) *The characteristic polynomial $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ of $\|\mathcal{A}\|$ satisfies either $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k$ for some $k \in \mathbb{N} \setminus \{0\}$, or $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k f(z)$ for $k \in \mathbb{N}$ and a monic polynomial $f(z) \in \mathbb{Q}[z]$ of degree $d > 0$ dividing $z^p - b$ for some $p \in \mathbb{N} \setminus \{0\}$ and $b \in \mathbb{N} \setminus \{0\}$.*

If in addition $\|\mathcal{A}\| \in \mathbb{N}\langle\langle c^* \rangle\rangle$, then the above statements are equivalent to determinisability of $\mathcal{A}$ over $\mathbb{N}$.

Proof. If $\mathcal{A}$ is determinisable, then there exists an accessible deterministic weighted automaton $\mathcal{B}$ over $\mathbb{Z}$ and $\Sigma = \{c\}$ such that $\|\mathcal{B}\| = \|\mathcal{A}\|$. Then Proposition 5.1 gives us either $\mathrm{ch}_{\mathcal{B}}(z) = z^q$ for some $q \in \mathbb{N} \setminus \{0\}$, or $\mathrm{ch}_{\mathcal{B}}(z) = z^m(z^p - b)$ for some $m \in \mathbb{N}$, $p \in \mathbb{N} \setminus \{0\}$, and $b \in \mathbb{Z} \setminus \{0\}$, while $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ divides $\mathrm{ch}_{\mathcal{B}}(z)$ by Corollary 4.5. This proves (*ii*). Moreover, we have already observed that (*ii*) implies (*iii*) by means of $(z^p - b)(z^p + b) = z^{2p} - b^2$.

Finally, let us assume (*iii*) and construct a deterministic weighted automaton $\mathcal{C}$ over $\mathbb{Q}$ and $\Sigma = \{c\}$ equivalent to $\mathcal{A}$ using Proposition 6.1. Let $\mathcal{P}_{\mathcal{C}} = (\ell, \mathbf{i}, \mu, \mathbf{f})$. Then it is clear that the entries of $\mathbf{i}$ and $\mu(c)$ are all in $\mathbb{N}$ and that every entry of $\mathbf{f}$ is equal to a coefficient of some word in $\|\mathcal{C}\| = \|\mathcal{A}\|$. Thus, $\|\mathcal{A}\| \in \mathbb{Z}\langle\langle c^* \rangle\rangle$ implies that the entries of $\mathbf{f}$ are in $\mathbb{Z}$, so that $\mathcal{C}$ is an automaton over $\mathbb{Z}$ and $\mathcal{A}$ is determinisable over $\mathbb{Z}$. Similarly, if $\|\mathcal{A}\| \in \mathbb{N}\langle\langle c^* \rangle\rangle$, then the entries of $\mathbf{f}$ have to be in $\mathbb{N}$, and $\mathcal{A}$ is determinisable over $\mathbb{N}$. ☐

We can now prove a Fatou-like property, which trivially implies that determinisation of unary weighted automata over $\mathbb{Z}$ and $\mathbb{N}$ is decidable. Note that the property for $\mathbb{N}$ holds although $\mathbb{Z}$ is not a Fatou extension of $\mathbb{N}$, *i.e.*, there are series with nonnegative integer coefficients that are rational over $\mathbb{Z}$, but not rational over $\mathbb{N}$ [8, Example 7.2.1] – even over a unary alphabet [8, Exercise 8.1.3].

Theorem 6.6. *A unary weighted automaton over $\mathbb{Z}$ (over $\mathbb{N}$) is determinisable over $\mathbb{Z}$ (over $\mathbb{N}$) if and only if it is determinisable over $\mathbb{Q}$.*

Proof. Let $\mathcal{A}$ be a unary weighted automaton over $\mathbb{Z}$ determinisable over $\mathbb{Q}$. It is well known that every weighted automaton over $\mathbb{Z}$ admits an equivalent minimal automaton over $\mathbb{Q}$ that is at the same time an automaton over $\mathbb{Z}$ [8, Section 7.1]. This means that $\mathrm{ch}_{\|\mathcal{A}\|}(z) \in \mathbb{Z}[z]$. At the same time, we have observed in our discussion following Theorem 6.2 that if $\mathcal{A}$ with $\|\mathcal{A}\| \neq 0$ is determinisable over $\mathbb{Q}$ and $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k f(z)$ as above, then $f(z)$ has to divide, for some $p \in \mathbb{N} \setminus \{0\}$, the polynomial $z^p - b$, where b is a power of $|a_0|$ for a_0 being the constant coefficient of $f(z)$. Thus $b \in \mathbb{N}$ and $\mathcal{A}$ is determinisable over $\mathbb{Z}$ by Proposition 6.5. The same result easily follows when $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k$ for some $k \in \mathbb{N} \setminus \{0\}$. If moreover $\mathcal{A}$ is an automaton over $\mathbb{N}$, then it is determinisable over $\mathbb{N}$, again by Proposition 6.5. □

The following corollary of Theorem 6.4 and Theorem 6.6 summarises our findings using the terminology of rational power series.

Corollary 6.7. *The sequentiality problem is decidable for univariate rational series over $\mathbb{Q}$, $\mathbb{Z}$, and $\mathbb{N}$.*

Remark 6.8. Recall that a weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ is *crisp-deterministic* [15, 16, 23] if it is deterministic and at the same time, the outputs of the transition weighting function σ and the initial weighting function ι are limited to be contained in $\{0, 1\}$.[18] It is not hard to see – and essentially already known [15, 34] – that a rational series $r \in \mathbb{F}\langle\langle\Sigma^*\rangle\rangle$, for some field $\mathbb{F}$ and alphabet Σ, is realised by a crisp-deterministic automaton if and only if it is of finite image, *i.e.*, if the set $\mathrm{Im}(r) = \{(r, w) \mid w \in \Sigma^*\}$ is finite; see also [15, Theorem 7.4] for a similar result in a more general setting of strong bimonoids. Indeed, $\|\mathcal{A}\|$ is obviously of finite image for a crisp-deterministic automaton $\mathcal{A}$. Conversely, every rational series r with finite image is known to be sequential [34, Proposition 12]. Then, given an accessible deterministic weighted automaton $\mathcal{A}$ such that $\|\mathcal{A}\| = r \neq 0$, it is straightforward to prove that the set of values $\sigma(\gamma)$ of runs γ from the unique initial state to q is finite for each state q of $\mathcal{A}$. One can thus change all (nonzero) transition weights to 1, while taking account of the original value of a run using a finite number of states, and incorporating these values into terminal weights.

The finite image property of rational series over $\mathbb{Q}$ is well known to be decidable via reduction to the finiteness problem for matrix semigroups [25, 35]; see also [8, Section 9.1] and [24, Section 5].

Now, using similar reasoning as in the proof of Proposition 6.5, it is easy to show that a *unary* weighted automaton $\mathcal{A}$ over $\mathbb{Q}$ and $\Sigma = \{c\}$ with $\|\mathcal{A}\| \neq 0$ admits a crisp-deterministic equivalent if and only if either $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k$ for some $k \in \mathbb{N} \setminus \{0\}$, or $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k f(z)$ for $k \in \mathbb{N}$ and a monic polynomial $f(z) \in \mathbb{Q}[z]$ of degree $d > 0$ dividing $z^p - 1$ for some $p \in \mathbb{N} \setminus \{0\}$. The latter happens if and only if $f(z)$ is a product of distinct cyclotomic polynomials, which can be easily checked (see the analysis following Algorithm 6.3). This observation implies a new polynomial-time algorithm for deciding the finite image property of *univariate* rational series over $\mathbb{Q}$.

Next, let us observe that Algorithm 6.3 can be easily modified so that a deterministic equivalent of its input automaton $\mathcal{A}$ is constructed whenever $\mathcal{A}$ is determinisable. Indeed, if it is found out that $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k$ for some $k \in \mathbb{N} \setminus \{0\}$ in Step 3, then a deterministic equivalent can be constructed as in the case (*i*) of Proposition 6.1. Otherwise, it follows from our analysis of Step 7 that it can be not only decided there whether $\mathrm{ch}_A(z)$ is a product of cyclotomic polynomials, but the cyclotomic factors $\Phi_{p_1}, \ldots, \Phi_{p_s}$ of $\mathrm{ch}_A(z)$ can actually be determined. If $\mathcal{A}$ is found to be determinisable, then the analysis of Algorithm 6.3 implies that $\mathrm{ch}_{\|\mathcal{A}\|}(z) = z^k f(z)$ for $k \in \mathbb{N}$ and a monic polynomial $f(z) \in \mathbb{Q}[z]$ of degree $d > 0$, both found in Step 4, where $f(z)$ divides $z^p - |a_0|^{p/d}$ for $p = d \cdot \mathrm{lcm}(p_1, \ldots, p_s)$ and a_0 the constant coefficient of $f(z)$. Thus, $\mathrm{ch}_{\|\mathcal{A}\|}(z)$ divides $z^{k+p} - |a_0|^{p/d} z^k$ and a deterministic equivalent of $\mathcal{A}$ can be constructed as in the case (*ii*) of Proposition 6.1.

[18] The term "crisp-deterministic" comes from the theory of fuzzy languages [33], which can be viewed as formal power series over the semiring $([0, 1], \max, \min, 0, 1)$ [39]. Here, crisp-deterministic automata are deterministic weighted automata that use, possibly except in the final weights, exclusively the crisp values 0 and 1.

Corollary 6.9. *There is an algorithm that, given a determinisable unary weighted automaton $\mathcal{A}$ over $\mathbb{Q}$ and $\Sigma = \{c\}$, computes its deterministic equivalent.*

Still using the notation from above, the automaton constructed in this way (in the nontrivial latter case) has $k + p$ states, where p is at most $d \cdot g(p)$ for the *Landau's function* g [38] given for all $n \in \mathbb{N} \setminus \{0\}$ by

$$g(n) = \max\{\operatorname{lcm}(p_1, \ldots, p_s) \mid s \in \mathbb{N} \setminus \{0\};\ p_1, \ldots, p_s \in \mathbb{N} \setminus \{0\};\ p_1 + \ldots + p_s \leq n\} = e^{(1+o(1))\sqrt{n \ln n}}.$$

In other words, the state complexity of determinisation of unary weighted automata over $\mathbb{Q}$ is bounded from above by

$$\Theta(n \cdot g(n)) = e^{(1+o(1))\sqrt{n \ln n}}.$$

We now show that this construction cannot be performed significantly better in general, as at least $g(n-1)$ states are necessary for determinisation of unary weighted automata over $\mathbb{Q}$ or, in fact, over $\mathbb{Z}$. Note that $\Theta(g(n))$ is also a tight upper bound for the state complexity of determinisation of unary nondeterministic finite automata without weights by the well-known result of M. Chrobak [13, 14].

Proposition 6.10. *For all $n \geq 3$, there exists a determinisable unary weighted automaton $\mathcal{A}_n$ over $\mathbb{Z}$ and $\Sigma = \{c\}$ with n states such that the number of states of the smallest equivalent deterministic weighted automaton $\mathcal{B}_n$ over $\mathbb{Q}$ and $\Sigma = \{c\}$ is at least $g(n-1)$.*

Proof. Given $n \geq 3$, let $s \in \mathbb{N} \setminus \{0\}$ and $p_1, \ldots, p_s \in \mathbb{N} \setminus \{0\}$ with $p_1 + \ldots + p_s \leq n - 1$ be such that $\operatorname{lcm}(p_1, \ldots, p_s) = g(n-1)$. Without loss of generality, let us assume that $p_1, \ldots, p_s \geq 2$ and that these numbers are pairwise distinct. Let $k = n - (1 + p_1 + \ldots + p_s)$. Then, define $\mathcal{A}_n$ by its associated linear representation $\mathcal{P}_{\mathcal{A}_n} = (n, \mathbf{i}, \mu, \mathbf{f})$, where $\mathbf{i} = (1, 0, \ldots, 0)$, $\mu(c)$ is the companion matrix of the polynomial $z^k \Phi_1(z) \Phi_{p_1}(z) \Phi_{p_2}(z) \ldots \Phi_{p_s}(z)$, and $\mathbf{f} = (0, \ldots, 0, 1)^T$. The automaton $\mathcal{A}_n$ is minimal, as the left quotients of $\|\mathcal{A}_n\|$ by $\varepsilon, c, \ldots, c^{n-1}$ are obviously linearly independent. Thus

$$\operatorname{ch}_{\|\mathcal{A}_n\|}(z) = \operatorname{ch}_{\mu(c)}(z) = z^k \Phi_1(z) \Phi_{p_1}(z) \Phi_{p_2}(z) \ldots \Phi_{p_s}(z).$$

Let $f(z) = \Phi_1(z) \Phi_{p_1}(z) \Phi_{p_2}(z) \ldots \Phi_{p_s}(z)$. As $\Phi_1(z) = z - 1$ is a factor of $f(z)$, it follows that $b = 1$ whenever $f(z)$ divides $z^p - b$ for some $p \in \mathbb{N} \setminus \{0\}$ and $b \in \mathbb{Q}$. Now, the smallest $p \in \mathbb{N} \setminus \{0\}$ such that $f(z)$ divides $z^p - 1$ is $\operatorname{lcm}(1, p_1, \ldots, p_s) = \operatorname{lcm}(p_1, \ldots, p_s) = g(n-1)$, by Theorem 2.9 and irreducibility of the cyclotomic polynomials. Thus, $g(n-1) + k$ states for $\mathcal{B}_n$ are necessary by Corollary 4.5 and Proposition 5.1, while they are also sufficient by Proposition 6.1. $\square$

7. Examples

We now illustrate the algorithms for deciding determinisability and actual determinisation, developed in the previous section, on a few simple examples.

Example 7.1. For the first example, consider the unary weighted automaton $\mathcal{A}_1$ over $\mathbb{Z}$ and $\Sigma = \{c\}$ depicted in Fig. 2. This automaton is clearly already minimal and the single matrix $\mu_1(c)$ of the linear representation $\mathcal{P}_{\mathcal{A}_1} = (n_1, \mathbf{i}_1, \mu_1, \mathbf{f}_1)$,

$$\mu_1(c) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -2 & 2 \end{pmatrix},$$

is the companion matrix of the characteristic polynomial $\operatorname{ch}_{\mathcal{A}_1}(z) = \operatorname{ch}_{\|\mathcal{A}_1\|}(z) = z^3 - 2z^2 + 2z$. There is thus no need to perform the minimisation algorithm.

The polynomial $\operatorname{ch}_{\|\mathcal{A}_1\|}(z)$ factorises as $\operatorname{ch}_{\|\mathcal{A}_1\|}(z) = z^k f(z)$, where $k = 1$ and $f(z) = z^2 - 2z + 2$; the polynomial $f(z)$ has degree $d = 2$ and constant coefficient $a_0 = 2$. The companion matrix C of $f(z)$ thus is

$$C = \begin{pmatrix} 0 & 1 \\ -2 & 2 \end{pmatrix},$$

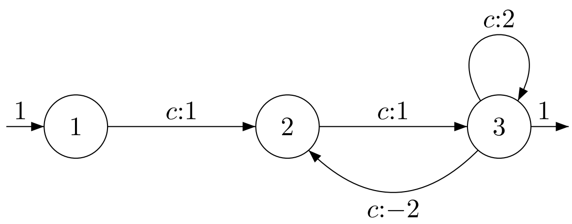


Figure 2: The automaton $\mathcal{A}_1$.

so that

$$A = \frac{1}{|a_0|} C^d = \frac{1}{2} C^2 = \begin{pmatrix} -1 & 1 \\ -2 & 1 \end{pmatrix}.$$

For the characteristic polynomial of the matrix A, we find out that

$$\mathrm{ch}_A(z) = z^2 + 1 = \Phi_4(z). \tag{14}$$

Moreover, by calculating the greatest common divisor of $f(z)$ and $f'(z) = 2z - 2$, we see that the roots of $f(z)$ are pairwise distinct. These observations imply that the automaton $\mathcal{A}_1$ is determinisable.

Let us also construct the deterministic weighted automaton over $\mathbb{Z}$ equivalent to $\mathcal{A}_1$. Using notation from the discussion preceding Algorithm 6.3, it follows by (14) that $p = 4 \cdot d = 4 \cdot 2 = 8$, which also implies $b = 2^4 = 16$. It follows that $f(z)$ divides $z^8 - 16$, which can, of course, be directly checked. As a consequence, $\mathcal{A}_1$ is equivalent to a deterministic weighted automaton $\mathcal{A}'_1$ such that

$$\mathcal{P}_{\mathcal{A}'_1} = (n'_1, \mathbf{i}'_1, \mu'_1, \mathbf{f}'_1),$$

where $n'_1 = 9$,

$$\mathbf{i}'_1 = (1, 0, 0, 0, 0, 0, 0, 0, 0),$$

$\mu'_1(c)$ is the companion matrix of $z(z^8 - 16) = z^9 - 16z$, *i.e.*,

$$\mu'_1(c) = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 16 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

and

$$\mathbf{f}'_1 = \big((\|\mathcal{A}_1\|, \varepsilon), (\|\mathcal{A}_1\|, c), (\|\mathcal{A}_1\|, c^2), (\|\mathcal{A}_1\|, c^3), (\|\mathcal{A}_1\|, c^4), (\|\mathcal{A}_1\|, c^5), (\|\mathcal{A}_1\|, c^6), (\|\mathcal{A}_1\|, c^7), (\|\mathcal{A}_1\|, c^8)\big)^T = \\ = (0, 0, 1, 2, 2, 0, -4, -8, -8)^T.$$

Note also that five states would in fact be sufficient; this is a consequence of the fact that we have limited ourselves to the case when $b > 0$. Abandoning this requirement, we find out that $f(z)$ also divides $z^4 + 4$, hence an automaton with five states can be constructed as in Proposition 6.1.

Example 7.2. Let us next consider a unary weighted automaton $\mathcal{A}_2$ over $\mathbb{Q}$ and $\Sigma = \{c\}$ with the corresponding linear representation given by $\mathcal{P}_{\mathcal{A}_2} = (n_2, \mathbf{i}_2, \mu_2, \mathbf{f}_2)$, where $n_2 = 4$, $\mathbf{i}_2 = (2/5, 0, 0, 1/5)$,

$$\mu_2(c) = \begin{pmatrix} -4/5 & 2 & 0 & -2/5 \\ -2/5 & 1 & 1/3 & -1/5 \\ -54/5 & -54 & -6 & -27/5 \\ -2/5 & 1 & 0 & -1/5 \end{pmatrix},$$

and $\mathbf{f}_2 = (0, 0, 3, 0)^T$. The minimal automaton $\mathcal{B}_2$ equivalent to $\mathcal{A}_2$ then has $\mathcal{P}_{\mathcal{B}_2} = (m_2, \mathbf{j}_2, \nu_2, \mathbf{g}_2)$, where $m_2 = 3$, $\mathbf{j}_2 = (1, 0, 0)$,

$$\nu_2(c) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -27 & -18 & -6 \end{pmatrix},$$

and $\mathbf{g}_2 = (0, 0, 1)^T$. The automaton $\mathcal{B}_2$ is depicted in Fig. 3.

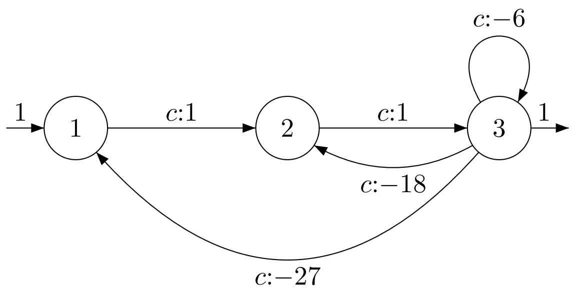


Figure 3: The automaton $\mathcal{B}_2$.

Hence, we get $\mathrm{ch}_{\|\mathcal{A}_2\|}(z) = \mathrm{ch}_{\mathcal{B}_2}(z) = z^3 + 6z^2 + 18z + 27 = f(z)$. The companion matrix C of $f(z)$ is given directly by $\nu_2(c)$, while

$$A = \frac{1}{27}C^3 = \begin{pmatrix} -1 & -2/3 & -2/9 \\ 6 & 3 & 2/3 \\ -18 & -6 & -1 \end{pmatrix}.$$

The characteristic polynomial of A is $\mathrm{ch}_A(z) = z^3 - z^2 - z + 1 = (z-1)^2(z+1) = \Phi_1(z)^2\Phi_2(z)$. Moreover, $f(z)$ is easily found out to have all roots simple. The automaton $\mathcal{A}_2$ is determinisable as a result. Finally, similarly as in the previous example, it is straightforward to observe that the automaton $\mathcal{A}_2$ admits a deterministic equivalent with six states.

Example 7.3. As a final example, consider the unary weighted automaton $\mathcal{A}_3$ over $\mathbb{N}$ and $\Sigma = \{c\}$ depicted in Fig. 4. That is, $\mathcal{P}_{\mathcal{A}_3} = (n_3, \mathbf{i}_3, \mu_3, \mathbf{f}_3)$ with $n_3 = 3$, $\mathbf{i}_3 = (1, 0, 0)$,

$$\mu_3(c) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix},$$

and $\mathbf{f}_3 = (0, 0, 1)^T$. This automaton is clearly minimal, while $\mu_3(c)$ is the companion matrix of the characteristic polynomial $\mathrm{ch}_{\|\mathcal{A}_3\|}(z) = \mathrm{ch}_{\mathcal{A}_3}(z) = z^3 - z - 1$. There is thus no need to perform minimisation.

Now, $C = \mu_3(c)$ and

$$A = -C^3 = \begin{pmatrix} -1 & -1 & 0 \\ 0 & -1 & -1 \\ -1 & -1 & -1 \end{pmatrix}.$$

Computing the characteristic polynomial of A gives us $\mathrm{ch}_A(z) = z^3 + 3z^2 + 2z + 1$. However, here we find that $\mathrm{ch}_A(z)$ is not divisible by any cyclotomic polynomial. We may thus conclude that the automaton $\mathcal{A}_3$ is not determinisable.

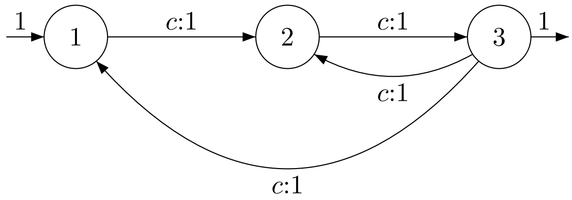


Figure 4: The automaton $\mathcal{A}_3$.

8. Conclusions

We have proved that the determinisability problem for unary weighted automata over $\mathbb{Q}$, $\mathbb{Z}$, and $\mathbb{N}$ is decidable, thus confirming in part a conjecture of S. Lombardy and J. Sakarovitch [34, Problem 1]. Algorithm 6.3 that we have proposed for this task uses at most $O(n^3)$ arithmetic operations over $\mathbb{Q}$ in each its run, where n is the number of states of the input automaton. The determinisability problem is thus decidable in polynomial time. Moreover, the algorithm for deciding determinisability can be easily modified so that a deterministic equivalent of the input automaton is produced on output when possible.

Besides the main results summarised above, we have observed that crisp-determinisability of weighted automata over $\mathbb{Q}$ can be decided using similar ideas as well. This implies an alternative algorithm for deciding the finite image property for univariate rational series over $\mathbb{Q}$. We have also seen in Proposition 6.10 that the upper bound on the state complexity of determinisation of unary weighted automata over $\mathbb{Q}$, implied by the determinisation algorithm described, is "almost tight".

Two obvious problems remain open, namely the remaining part of [34, Problem 1] – *i.e.*, the decidability of determinisability for weighted automata over $\mathbb{Q}$ and larger than unary alphabets – and the possible extension of the results of this article to other fields. In addition, findings related to Proposition 6.10 suggest the possibility of studying descriptional complexity of weighted automata over fields in greater depth.

References

[1] C. Allauzen and M. Mohri. Efficient algorithms for testing the twins property. *Journal of Automata, Languages and Combinatorics*, 8(2):117–144, 2003.

[2] J. Alman and V. V. Williams. A refined laser method and faster matrix multiplication. In *Symposium on Discrete Algorithms, SODA 2021*, pages 522–539. SIAM, 2021.

[3] A. Arnold and M. Monagan. Calculating cyclotomic polynomials. *Mathematics of Computation*, 80(276):2359–2379, 2011.

[4] E. Bach and J. Shallit. *Algorithmic Number Theory. Volume I: Efficient Algorithms.* The MIT Press, 1996.

[5] M.-P. Béal, S. Lombardy, and J. Sakarovitch. On the equivalence of $\mathbb{Z}$-automata. In *Automata, Languages and Programming, ICALP 2005*, pages 397–409. Springer, 2005.

[6] M.-P. Béal, S. Lombardy, and J. Sakarovitch. Conjugacy and equivalence of weighted automata and functional transducers. In *Computer Science – Theory and Applications, CSR 2006*, pages 58–69. Springer, 2006.

[7] J. Bell and D. Smertnig. Noncommutative rational Pólya series. *Selecta Mathematica*, 27(3):article 34, 2021.

[8] J. Berstel and C. Reutenauer. *Noncommutative Rational Series with Applications.* Cambridge University Press, 2011.

[9] F. Beukers and C. J. Smyth. Cyclotomic points on curves. In *Number Theory for the Millennium I*, pages 67–85. A K Peters, 2002.

[10] R. E. Blahut. *Algebraic Codes for Data Transmission.* Cambridge University Press, Cambridge, 2003.

[11] R. J. Bradford and J. H. Davenport. Effective tests for cyclotomic polynomials. In *Symbolic and Algebraic Computation, ISSAC 1988*, pages 244–251, 1989.

[12] A. Cardon and M. Crochemore. Détermination de la représentation standard d'une série reconnaissable. *Informatique Théorique et Applications*, 14(4):371–379, 1980.

[13] M. Chrobak. Finite automata and unary languages. *Theoretical Computer Science*, 47:149–158, 1986.

[14] M. Chrobak. Errata to: "Finite automata and unary languages" [Theoret. Comput. Sci. 47 (1986) 149–158]. *Theoretical Computer Science*, 302(1–3):497–498, 2003.

[15] M. Ćirić, M. Droste, J. Ignjatović, and H. Vogler. Determinization of weighted finite automata over strong bimonoids. *Information Sciences*, 180:3497–3520, 2010.

[16] M. Droste, Z. Fülöp, D. Kószó, and H. Vogler. Crisp-determinization of weighted tree automata over additively locally finite and past-finite monotonic strong bimonoids is decidable. In *Descriptional Complexity of Formal Systems, DCFS 2020*, pages 39–51. Springer, 2020.

[17] M. Droste and W. Kuich. Semirings and formal power series. In M. Droste, W. Kuich, and H. Vogler, editors, *Handbook of Weighted Automata*, chapter 1, pages 3–28. Springer, 2009.

[18] M. Droste, W. Kuich, and H. Vogler, editors. *Handbook of Weighted Automata*. Springer, 2009.

[19] M. Droste and D. Kuske. Weighted automata. In J.-É. Pin, editor, *Handbook of Automata Theory, Vol. 1*, chapter 4, pages 113–150. European Mathematical Society, 2021.

[20] J.-G. Dumas, C. Pernet, and Z. Wan. Efficient computation of the characteristic polynomial. In *Symbolic and Algebraic Computation, ISSAC 2005*, pages 140–147. ACM, 2005.

[21] D. S. Dummit and R. M. Foote. *Abstract Algebra*. John Wiley & Sons, 3rd edition, 2004.

[22] S. Elaydi. *An Introduction to Difference Equations*. Springer, 3rd edition, 2005.

[23] Z. Fülöp, D. Kószó, and H. Vogler. Crisp-determinization of weighted tree automata over strong bimonoids. *Discrete Mathematics and Theoretical Computer Science*, 23(1):article 16, 2021.

[24] V. Halava. Decidable and undecidable problems in matrix theory. Technical report, Turku Centre for Computer Science, 1997. Available at `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.31.5792&rep=rep1&type=pdf`.

[25] G. Jacob. La finitude des représentations linéaires des semi-groupes est décidable. *Journal of Algebra*, 52(2):437–459, 1978.

[26] G. Jacob, C. Reutenauer, and J. Sakarovitch. On a divisibility property of Fibonacci polynomials. Available at `https://perso.telecom-paristech.fr/jsaka/PUB/Files/DPFP.pdf`, 2006.

[27] D. Kalman. The generalized Vandermonde matrix. *Mathematics Magazine*, 57(1):15–21, 1984.

[28] W. Keller-Gehrig. Fast algorithms for the characteristic polynomial. *Theoretical Computer Science*, 36:309–317, 1985.

[29] W. G. Kelley and A. C. Peterson. *Difference Equations*. Academic Press, 2nd edition, 2001.

[30] S. Kiefer. Notes on equivalence and minimization of weighted automata. Available at `https://arxiv.org/abs/2009.01217`, 2020.

[31] D. Kirsten and S. Lombardy. Deciding unambiguity and sequentiality of polynomially ambiguous min-plus automata. In *Symposium on Theoretical Aspects of Computer Science, STACS 2009*, pages 589–600, 2009.

[32] D. Kirsten and I. Mäurer. On the determinization of weighted automata. *Journal of Automata, Languages and Combinatorics*, 10(2–3):287–312, 2005.

[33] E. T. Lee and L. A. Zadeh. Note on fuzzy languages. *Information Sciences*, 1(4):421–434, 1969.

[34] S. Lombardy and J. Sakarovitch. Sequential? *Theoretical Computer Science*, 356:224–244, 2006.

[35] A. Mandel and I. Simon. On finite semigroups of matrices. *Theoretical Computer Science*, 5(2):101–111, 1977.

[36] M. Mohri. Finite-state transducers in language and speech processing. *Computational Linguistics*, 23(2):269–311, 1997.

[37] M. Mohri. Weighted automata algorithms. In M. Droste, W. Kuich, and H. Vogler, editors, *Handbook of Weighted Automata*, chapter 6, pages 213–256. Springer, 2009.

[38] J.-L. Nicolas. On Landau's function $g(n)$. In R. L. Graham and J. Nešetřil, editors, *The Mathematics of Paul Erdös I*, pages 228–240. Springer, 1997.

[39] G. Rahonis. Fuzzy languages. In M. Droste, W. Kuich, and H. Vogler, editors, *Handbook of Weighted Automata*, chapter 12, pages 481–517. Springer, 2009.

[40] J. B. Rosser and L. Schoenfeld. Approximate formulas for some functions of prime numbers. *Illinois Journal of Mathematics*, 6(1):64–94, 1962.

[41] J. Sakarovitch. *Elements of Automata Theory*. Cambridge University Press, Cambridge, 2009.

[42] J. Sakarovitch. Rational and recognisable power series. In M. Droste, W. Kuich, and H. Vogler, editors, *Handbook of Weighted Automata*, chapter 4, pages 105–174. Springer, 2009.

[43] M.-P. Schützenberger. On the definition of a family of automata. *Information and Control*, 4(2–3):245–270, 1961.

[44] V. Strassen. Gaussian elimination is not optimal. *Numerische Mathematik*, 13(4):354–356, 1969.

[45] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, Cambridge, 3rd edition, 2013.