

Functional Closure Properties of Finite $\mathbb{N}$ -weighted Automata

Julian Dörfler  

Saarland Informatics Campus (SIC), Saarbrücken Graduate School of Computer Science, Saarland University, Germany

Christian Ikenmeyer  

University of Warwick, United Kingdom

Abstract

We determine all functional closure properties of finite $\mathbb{N}$ -weighted automata, even all multivariate ones, and in particular all multivariate polynomials. We also determine all univariate closure properties in the promise setting, and all multivariate closure properties under certain assumptions on the promise, in particular we determine all multivariate closure properties where the output vector lies on a monotone algebraic graph variety.

2012 ACM Subject Classification Theory of computation $\rightarrow$ Automata extensions

Keywords and phrases Finite automata, weighted automata, counting, closure properties, algebraic varieties

Funding *Christian Ikenmeyer*: EPSRC EP/W014882/2

1 Finite $\mathbb{N}$ -weighted automata and functional closure properties

Let Σ be a finite set, for example $\Sigma = \{0, 1\}$. A finite $\mathbb{N}$ -weighted automaton with all weights 1 is a nondeterministic finite automaton that on input $w \in \Sigma^*$ outputs the number of accepting computation paths on input w , instead of just outputting whether or not an accepting computation path exists, see Def. 2.1 for the formal definition¹. While every nondeterministic finite automaton determines a subset of Σ^* , a finite $\mathbb{N}$ -weighted automaton computes a function $\Sigma^* \rightarrow \mathbb{N}$. A function $f : \Sigma^* \rightarrow \mathbb{N}$ can be presented as the series $\sum_{w \in \Sigma^*} f(w)w$, and the set of series is denoted by $\mathbb{N}\langle\langle \Sigma^* \rangle\rangle$ in the automata literature, see e.g. [10]². The natural way of adding two functions $\Sigma^* \rightarrow \mathbb{N}$ and adding two series in $\mathbb{N}\langle\langle \Sigma^* \rangle\rangle$ coincides, but in both presentations we have a natural way of taking the product, and those do not coincide:

1. Pointwise product of functions $\Sigma^* \rightarrow \mathbb{N}$. This is called the Hadamard product.
2. Convolution of series, called the Cauchy product.

A series f is called *recognizable* if there is a finite $\mathbb{N}$ -weighted automaton that computes f . The set of recognizable series is denoted by $\mathbb{N}^{\text{rec}}\langle\langle \Sigma^* \rangle\rangle$ in [10], but we denote it by $\#FA$, to emphasise that we undertake a study similar to $\#P$ in [13, Thm 3.13], [5, Thm 6], and recently [14], but instead of polynomial-time Turing machines we study finite automata. The Kleene-Schützenberger theorem states that $\#FA$ is the smallest set that contains all support 1 series and is closed under sums, Cauchy products, and Kleene-iterations (whenever well-defined, a Kleene-iteration is the sum of all Cauchy powers), see [10, §4], but we will not need this insight.

In this paper we study the functional closure properties³ of $\#FA$. A function $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ is called a *functional closure property* of $\#FA$ if for all $f_1 \in \#FA$, $f_2 \in \#FA$, $\dots$, $f_m \in \#FA$ we have that $\varphi(f_1, \dots, f_m) \in \#FA$. By $\varphi(f_1, \dots, f_m)$ we mean the function that on input $w \in \Sigma^*$ outputs $\varphi(f_1(w), \dots, f_m(w))$.

Classically, one of the simplest functional closure properties of $\#FA$ is $\varphi : \mathbb{N}^2 \rightarrow \mathbb{N}$, $\varphi(f_1, f_2) = f_1 + f_2$. This is a functional closure property of $\#FA$, because given $f_1 \in \#FA$ and $f_2 \in \#FA$, we can show $\varphi(f_1, f_2) = f_1 + f_2 \in \#FA$ by an easy construction: The new NFA consists of a copy of the NFA for f_1 and a copy of the NFA for f_2 , and makes an initial nondeterministic choice as to which NFA to run, see Lemma 3.1 for the details.

Another classical simple functional closure property of $\#FA$ is $\varphi : \mathbb{N}^2 \rightarrow \mathbb{N}$, $\varphi(f_1, f_2) = f_1 \cdot f_2$. This corresponds to the Hadamard product. This is a functional closure property of $\#FA$, because given $f_1 \in \#FA$ and $f_2 \in \#FA$, we can show $\varphi(f_1, f_2) = f_1 \cdot f_2 \in \#FA$ by the following construction: The new NFA consists of the product NFA of the NFAs for f_1 and f_2 , and the accepting states correspond to pairs of accepting states, see Lemma 3.2 for the details. This product construction corresponds to the Hadamard product.

The Cauchy product is also a product on the set $\#FA$, but we explain now that the Cauchy product is not “functional”, and hence it is out of scope for this type of studies. If $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ is a functional closure property of $\#FA$, then we can study the corresponding

¹ We use the equality of the number of accepting paths of an NFA and the output of the corresponding $\mathbb{N}$ -weighted automaton, see [10, Exa. 2.2].

² A series with finite support is called a *polynomial*, but we will not be concerned with the support of series in this paper. Instead, we use the term *polynomial* as it is used in commutative algebra, and we mean multivariate polynomials with rational coefficients.

³ See [13, Sec. 1] for the naming *functional closure property*. A different reasonable name would be *pointwise closure property*.

map $\tilde{\varphi} : \underbrace{\#FA \times \#FA \times \cdots \times \#FA}_{m \text{ times}} \rightarrow \#FA$. Observe that if φ is a functional closure property

of $\#FA$, then by definition we have that for all pairs $(w, w') \in \Sigma^* \times \Sigma^*$:

if $(f_1(w), \dots, f_m(w)) = (f_1(w'), \dots, f_m(w'))$, then $\tilde{\varphi}(f_1, \dots, f_m)(w) = \tilde{\varphi}(f_1, \dots, f_m)(w')$.

Let $\zeta : \#FA \rightarrow \#FA$ denote the Cauchy square. We use the observation above to show that ζ is not equal to $\tilde{\varphi}$ for any $\varphi : \mathbb{N} \rightarrow \mathbb{N}$. Let $m = 1$ and $f(w) = 1$ if $w = 1$, $f(w) = 0$ otherwise. Clearly, $f \in \#FA$. Then $\zeta(f)(11) = 1$, and $\zeta(f)(w) = 0$ for all $w \neq 11$. In particular $\zeta(f)(0) \neq \zeta(f)(11)$, even though $f(0) = f(11)$. Hence, $\zeta \neq \tilde{\varphi}$ for all $\varphi : \mathbb{N} \rightarrow \mathbb{N}$.

Numerous functional closure properties of $\#FA$ exist, for example the safe decrementation $\max\{0, f_1 - 1\}$, and the binomial coefficient $\binom{f_1}{2}$. But not all non-negative functions are functional closure properties of $\#FA$, for example $(f_1 - f_2)^2$ is not, which can be shown using the Pumping Lemma. In this paper, we determine *all* functional closure properties of $\#FA$, see §1.2 for the detailed statement.

1.1 Motivation

Functional closure properties can be studied for many different counting machine models (also for example with different types of oracle access) and different types of input sets. The first study of this type was done for nondeterministic polynomial-time Turing machines, i.e., the class $\#P$, see [13], [5], and the recent [14]. Recall that the class $\#P$ is the class of functions $f : \Sigma^* \rightarrow \mathbb{N}$ for which a nondeterministic polynomial time Turing machine M exists such that for all $w \in \Sigma^*$ the number of accepting paths for the computation $M(w)$ is exactly $f(w)$. The papers mentioned above prove that the relativizing multivariate polynomial closure properties are exactly those polynomials that have nonnegative integers in their expansion over the binomial basis, see [14]. A functional closure property $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ of $\#P$ is *relativizing* if φ is a closure property for all $\#P^A$, where $A \subset \Sigma^*$ is some oracle. The hope is that for simpler models of computation no oracle access is required to determine the functional closure properties, and we show that this is true for $\#FA$, see §1.2.

Functional closure properties can be used directly to construct combinatorial proofs of equalities and inequalities. For example, Fermat's little theorem states that p divides $a^p - a$. The quantity $\frac{1}{p}(a^p - a)$ has a combinatorial interpretation, which can be deduced from the fact that $\frac{1}{p}((f_1)^p - f_1)$ is a univariate functional closure property of $\#P$, see [14, Prop. 7.3.1], which coincides with the original proof [20], see also [19, eq. (5)]. On the other hand, if a function is not a functional closure property, then this means in a very strong sense that there is no combinatorial interpretation for the quantity it describes. For example, the Hadamard inequality ([12, §2.13], [4, §2.11], [14, eq. (2)]) states that

$$\det \begin{pmatrix} a_{11} & \cdots & a_{1d} \\ \vdots & \ddots & \vdots \\ a_{d1} & \cdots & a_{dd} \end{pmatrix}^2 \leq \prod_{i=1}^d (a_{i1}^2 + \cdots + a_{id}^2).$$

One could try to prove this by finding a combinatorial interpretation of the difference $\mathcal{H} \geq 0$ of the right-hand side and the left-hand side, but even for $d = 3$ we have that

$$\varphi(f_1, \dots, f_9) = (f_1^2 + f_2^2 + f_3^2) \cdot (f_4^2 + f_5^2 + f_6^2) \cdot (f_7^2 + f_8^2 + f_9^2) - \det \begin{pmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & f_9 \end{pmatrix}^2$$

is not a 9-variate relativizing functional closure property of $\#P$, see [14, §7.2]. In particular, if the function is not a closure property of $\#P$, then there are instantiations $f_1, \dots, f_9 \in \#P$ such that $\varphi(f_1, \dots, f_9)$ is not in $\#P$, whereas a combinatorial interpretation of $\mathcal{H}$ should yield $\varphi(f_1, \dots, f_9) \in \#P$. This does not rule out a more indirect combinatorial proof for the inequality: For example, for proving combinatorially that $(a - 1)^2 \geq 0$ one could try to interpret the quantity $(a - 1)^2$ combinatorially, but $(f_1 - 1)^2$ is not a relativizing closure

property of $\#P$. However, $f_1 \cdot (f_1 - 1)^2 = 6\binom{f_1}{3} + 2\binom{f_1}{2}$ is a relativizing closure property of $\#P$ (see [14, §2.4]). There is an obvious combinatorial interpretation of $6\binom{a}{3} + 2\binom{a}{2}$ as counting size 2 and 3 subsets with multiplicity 6 and 2, respectively. Hence this gives an indirect combinatorial proof for the inequality $(a - 1)^2 \geq 0$ by providing a combinatorial interpretation for $a(a - 1)^2$.

Some inequalities are only true if the inputs satisfy certain constraints. For example, the *Ahlswede Daykin inequality*, see [1], [2], [14, §1.2(3)]: If $a_0b_0 \geq c_0d_0$ and $a_0b_1 \geq c_0d_1$ and $a_1b_0 \geq c_0d_1$ and $a_1b_1 \geq c_1d_1$, then $(c_0 + c_1)(d_0 + d_1) \geq (a_0 + a_1)(b_0 + b_1)$. If all quantities are in $\#P$, including the differences $c_0d_0 - a_0b_0$, can we conclude that $(c_0 + c_1)(d_0 + d_1) - (a_0 + a_1)(b_0 + b_1)$ is in $\#P$? This is an example of a promise problem: We are given twelve $\#P$ functions $a_0, a_1, b_0, b_1, c_0, c_1, d_0, d_1, h_1, h_2, h_3, h_4$ with the guarantee that $a_0b_0 + h_1 = c_0d_0$, $a_0b_1 + h_2 = c_0d_1$, $a_1b_0 + h_3 = c_0d_1$, $a_1b_1 + h_4 = c_1d_1$. In other words, the 12-dimensional output vector that we get for every $w \in \Sigma^*$ lies on a codimension 4 algebraic subvariety in $\mathbb{Q}^{12}$. Recall that an algebraic subvariety is defined as the simultaneous zero set of a set of polynomials. Since the 4 variables $h_1, \dots, h_4$ are determined by the other 8, this variety is a so-called *graph* or *graph variety*. Numerous questions about combinatorial proofs for inequalities from different areas of mathematics can be phrased in the language of graph varieties, see [14]. The idea is to collect the equations for a set S (the variety) into what is called the *vanishing ideal* I , i.e., $I = I(S) = \{\varphi \in \mathbb{Q}[f_1, \dots, f_m] \mid \forall (f_1, \dots, f_m) \in S : \varphi(f_1, \dots, f_m) = 0\}$; and define the *coordinate ring* $\mathbb{Q}[S]$ as the quotient ring $\mathbb{Q}[f_1, \dots, f_m]/I(S)$, see [8]. An element in the quotient ring is a coset with respect to the vanishing ideal. If there exists a representative φ' in a coset $\varphi + I$ that is a functional closure property of $\#P$, then every function in $\varphi + I$ is a promise closure property of $\#P$ on the variety S . It is desirable to also have the opposite direction, but this only holds under some reasonable restrictions on S , in particular it holds for all graph varieties. This is used in [14, Prop. 2.5.1] to show that $c_0d_0 - a_0b_0$ is not a relativizing promise closure property of $\#P$ on this graph variety. We prove the same strong dichotomy for monotone graph varieties for $\#FA$ instead of $\#P$, see Theorem 4.9.

The systematic study of combinatorial interpretations and combinatorial proofs via definitions from computational complexity theory is a very recent research direction [17, 18, 14, 19, 15, 6, 7]. The goal is to determine whether or not certain quantities admit a combinatorial description or not. Famous open questions of this type in algebraic combinatorics have been listed by Stanley in [24], for example his problems 9, 10, and 12. As many combinatorialists do, Stanley has phrased his questions in an informal way without mentioning counting classes.

The class $\#P$ is the correct class for some purposes, but for others it is too large. For example, the determinant of a skew-symmetric matrix with entries from $\{-1, 0, 1\}$ is always non-negative, but this quantity is trivially in $\#P$, because the determinant can be computed in polynomial time. This gives no satisfying insight into whether or not this quantity has a combinatorial interpretation. Smaller counting classes are required (see also the discussion in [17, §1]), and we provide the first study of functional closure properties for the subclass $\#FA \subset \#P$. Unlike the classification for $\#P$, our results do not rely on oracle separations, i.e., our classification is entirely unconditional.

1.2 Our results

Let $n \bmod p \in \{0, \dots, p-1\}$ denote the smallest nonnegative r such that $n \equiv_p r$. A function $\varphi : \mathbb{N} \rightarrow \mathbb{N}$ is called *ultimately PORC* (Polynomial On Residue Classes⁴) if $\exists p, N \in \mathbb{N}$ and there exist polynomials $\varphi_0, \dots, \varphi_{p-1} : \mathbb{N} \rightarrow \mathbb{Q}$ such that for every $n \geq N$ we have $\varphi(n) = \varphi_{n \bmod p}(n)$ ⁵.

We first classify the univariate functional closure properties of #FA:

- **Theorem** (see Theorem 3.22). *A function $\varphi : \mathbb{N} \rightarrow \mathbb{N}$ is a functional closure property of #FA if and only if φ is an ultimately PORC function.*

More generally, we classify the multivariate functional closure properties of #FA:

- **Theorem** (see Theorem 3.23). *A function $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ is a functional closure property of #FA if and only if φ can be written as a finite sum of finite products of univariate ultimately PORC functions.*

We analyze the special case of multivariate polynomials:

- **Theorem** (see Lemma 3.26). *A multivariate polynomial $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ with rational coefficients is a functional closure property of #FA iff for every ψ that can be formed from φ by replacing any subset of variables – including the empty set – by constants from $\mathbb{N}$, then all dominating terms of ψ in the binomial basis have positive coefficients.*

We lift this result to monotone graph varieties (and to more general sets, see Theorem 4.6), where we get exactly the desirable classification given by the vanishing ideal:

- **Theorem** (see Theorem 4.9). *Let S be a monotone graph variety and let $I = I(S)$ be its vanishing ideal. A multivariate polynomial $\varphi : S \rightarrow \mathbb{N}$ is a functional promise closure property of #FA with regard to S if and only if there exists $\psi \in I$ such that $\varphi + \psi$ is a multivariate functional closure property of #FA.*

2 Notation

Let $\mathbb{N} = \{0, 1, 2, \dots\}$. For a finite set Σ let Σ^* denote the set of all finite length sequences with elements from Σ . The vector space of multivariate polynomials $\mathbb{Q}[f_1, \dots, f_m]$ in variables $f_1, \dots, f_m$ has a basis given by products of binomial coefficients: $\{\prod_{i=1}^m \binom{f_i}{c_i}\}_{c_1, \dots, c_m}$, where each $c_i \in \mathbb{N}$. Here we used $\binom{x}{c} = \frac{1}{c!} x \cdot (x-1) \cdot \dots \cdot (x-c+1)$ as a polynomial. This is called the *binomial basis*. A multivariate polynomial φ is called *integer valued* if $\varphi(\mathbb{Z}^m) \subseteq \mathbb{Z}$, which is equivalent to $\varphi(\mathbb{N}^m) \subseteq \mathbb{Z}$, and which is also equivalent to all coefficients in the binomial basis being integers, see for example [14, Prop. 4.2.1] for a short proof of this classical fact.

We now recall (see [10, Def. 2.1]) our main model of computation, the finite $\mathbb{N}$ -weighted automaton, which we just call non-deterministic finite automaton (NFA) for brevity.

► **Definition 2.1.** *An NFA M is a tuple $(Q, \Sigma, \text{wt}, \text{in}, \text{out})$ where the set of states Q and the alphabet Σ are finite sets and $\text{wt} : Q \times \Sigma \times Q \rightarrow \mathbb{N}$ is the weighted transition function, $\text{in} : Q \rightarrow \mathbb{N}$ are the weighted initial states and $\text{out} : Q \rightarrow \mathbb{N}$ are the weighted accepting states⁶. A computation P for a word $w = w_1 \dots w_n \in \Sigma^*$ of length n is a sequence $q_0 q_1 \dots q_n$*

⁴ PORC functions are also known as quasipolynomials or pseudopolynomials, but we want to avoid those names for the potential confusion to quasipolynomial growth and pseudopolynomial running times.

⁵ Note that each φ in this paper is defined on the natural numbers and maps to the natural numbers, which is a subtle restriction. For example, a univariate polynomial $\varphi : \mathbb{Q} \rightarrow \mathbb{Q}$ maps integers to integers if and only if its coefficients in the binomial basis are integers, see Section 2. However, non-negativity is not an algebraic property. Also note that for the case of φ being just a univariate polynomial, the corresponding linear recursive sequence can have negative entries in the matrix.

⁶ Note that in definitions by other authors one can find simpler versions of NFAs, in particular unweighted initial and accepting states and unweighted edges while also restricting to a single initial state. We will

in Q^{n+1} . It has multiplicity or weight⁷ $\mathbf{w}(P) = \text{in}(q_0) \cdot \prod_{i=1}^n \text{wt}(q_{i-1}, w_i, q_i) \cdot \text{out}(q_n)$ and partial weight $\underline{\mathbf{w}}(P) = \text{in}(q_0) \cdot \prod_{i=1}^n \text{wt}(q_{i-1}, w_i, q_i)$. We say that M computes $f : \Sigma^* \rightarrow \mathbb{N}$ where $f(w)$ is the sum of the weights over all computations of M on w . The class $\#FA$ is defined as the set of all functions $f : \Sigma^* \rightarrow \mathbb{N}$ that are computed by NFAs.

If needed to distinguish these for different automata, we use a corresponding subscript, for example the weights of computations in M_f would be denoted by $\mathbf{w}_f$, etc.

► **Definition 2.2** (Simple NFA). We say an NFA $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ is simple if $\text{im wt}, \text{im in}, \text{im out} \subseteq \{0, 1\}$.

The notion of a simple NFA also motivates our use of the term NFA opposed to $\mathbb{N}$ -weighted automaton: We simply count the number of accepting paths of M on a word w . This is in line with $\#P$ counting the number of accepting paths on a polynomial time non-deterministic Turing machine.

► **Lemma 2.3** (Folklore). For every NFA M there exists a simple NFA M' computing the same function.

A proof of this simple fact can be found in the Appendix, for the sake of completeness. We denote by $a \equiv_p b$ that $a \in \mathbb{N}$ and $b \in \mathbb{N}$ are congruent modulo $p \in \mathbb{N} \setminus \{0\}$. The indicator function $\mathbb{1}_{n=c} : \mathbb{N} \rightarrow \mathbb{N}$ is defined as $n \mapsto \begin{cases} 1 & \text{if } n = c \\ 0 & \text{otherwise} \end{cases}$ and analogously for different conditions. By abuse of notation, if we have a function $f : \Sigma^* \rightarrow \mathbb{N}$ and an expression $\varphi : \mathbb{N} \rightarrow \mathbb{N}$ in n , we replace n by f in the expression to denote $\varphi \circ f$. For example we use $\mathbb{1}_{f=c}$ to denote the function $w \mapsto \mathbb{1}_{f(w)=c}$, similarly $\binom{f}{2}$ denotes the function $w \mapsto \binom{f(w)}{2}$. Furthermore we use the notation $[n]$ to denote the set $\{1, \dots, n\}$ for any $n \in \mathbb{N}$.

3 Functional closure properties

3.1 Univariate functional closure properties

We say a function $\varphi : \mathbb{N} \rightarrow \mathbb{N}$ is a functional closure property of $\#FA$ if $\varphi(\#FA) \subseteq \#FA$, i.e. if for every function $f \in \#FA$ the function $\varphi \circ f$ is also in $\#FA$. Our goal in this section is to classify all functional closure properties of $\#FA$. They will be precisely the ultimately PORC functions.

We call a function $\varphi : \mathbb{N} \rightarrow \mathbb{N}$ an *ultimately almost PORC function* if there is a *quasiperiod* p , an *offset* $N \in \mathbb{N}$ and *constituents* $\varphi_0, \dots, \varphi_{p-1} : \mathbb{N} \rightarrow \mathbb{Q}$, where each φ_i is either a polynomial with rational coefficients or a function in $2^{\Theta(n)}$, and for every $n \geq N$ we have $\varphi(n) = \varphi_{n \bmod p}(n)$. If all the constituents are polynomials, we call φ an *ultimately PORC function*. The smallest representative of each constituent and of the finite cases before the periodic behaviour is captured by the *shifted remainder* operator $n \bmod_p N$, defined via

$$n \bmod_p N = \begin{cases} n & \text{if } n < N \\ \min\{k \geq N \mid k \equiv_p n\} & \text{if } n \geq N \end{cases}$$

see soon that working with unweighted NFAs is not a restriction, but additionally restricting the model to have a single initial state is strictly weaker, since this model could not compute any function f with $f(\varepsilon) > 1$. To obtain the same expressiveness one would have to additionally allow for ε -transitions, while disallowing cycles of ε -transitions to prevent infinite values for f .

⁷ We will use both of these terms interchangeably. For a weighted automaton, calling this weight is more natural, while when looking at the underlying graph as a multigraph, multiplicity of paths and walks is more natural.

The first half of the section is dedicated to proving that every ultimately PORC function is a functional closure property of $\#FA$, see Lemma 3.18. In order to prove this we show that $\#FA$ is closed under

- **Lemma 3.1** (Addition). *If $f, g \in \#FA$, then $f + g \in \#FA$.*
- **Lemma 3.2** (Multiplication). *If $f, g \in \#FA$, then $f \cdot g \in \#FA$.*
- **Lemma 3.9** (Subtraction of constants). *If $f \in \#FA$, then $\forall c \in \mathbb{N} : \max(f - c, 0) \in \#FA$.*
- **Lemma 3.10** (Clamping). *If $f \in \#FA$, then $\min(f, c) \in \#FA$ for any constant $c \in \mathbb{N}$.*
- **Lemma 3.11** (Comparison with constants). *If $f \in \#FA$, then the functions $\mathbb{1}_{f=c}$, $\mathbb{1}_{f \leq c}$, $\mathbb{1}_{f \geq c}$ are in $\#FA$ for any constant $c \in \mathbb{N}$.*
- **Lemma 3.13** (Division by constants). *If $f \in \#FA$, then $\forall c \in \mathbb{N} \setminus \{0\} : \lfloor f/c \rfloor \in \#FA$.*
- **Lemma 3.14** (Modular arithmetic). *If $f \in \#FA$, then the function $\mathbb{1}_{f \equiv c \pmod d}$ is in $\#FA$ for any constants $c \in \mathbb{N} \setminus \{0\}$ and $d \in \mathbb{Z}_c$.*
- **Lemma 3.15** (Binomial coefficients). *If $f \in \#FA$, then $\binom{f}{c} \in \#FA$ for any constant $c \in \mathbb{N}$.*

While addition and multiplication are technically bivariate functional closure properties, we list them here already since they are abundantly used throughout the proofs of the univariate functional closure properties. Proofs of those two classical results can be found in [9][Ch. 4.1 and 4.2.2] and in the appendix, for the sake of completeness.

With the exception of binomial coefficients all the other closure properties need to be able to “remove” some of the possible computations. For example, consider decrementation, the special case of truncated subtraction by one, and consider some simple NFA M computing some strictly positive function f . We now want to construct an NFA M' that computes $f - 1$, i.e. an NFA that has exactly one non-zero computation less than M (assuming computations of weights zero or one). For this we want a procedure to single out one non-zero computation of M to then change its weight to zero. For stronger models of computation – like polynomial time non-deterministic Turing machines – this approach seems hopeless. Already deciding the existence of one such computation is NP-hard. However for NFAs, deciding the existence of a non-zero computation can be decided by a deterministic finite automaton, namely the powerset automaton. Adjusting the powerset construction to filter out a single non-zero computation, namely the lexicographically minimal one can then be used to show that decrementation is a closure property of $\#FA$.

Generalizing this approach to more general properties about the computations gives us the framework of stepwise computation properties:

► **Definition 3.3** (Stepwise computation property). *Let $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ be an NFA. A stepwise computation property prop is defined as $\text{prop} = (S, \text{init}, \text{step}, \text{cond})$ where S is a finite set and $\text{init} : Q \rightarrow S$, $\text{step} : Q \times \Sigma \times Q \times S \rightarrow S$ and $\text{cond} : S \rightarrow \{0, 1\}$ are functions. For $w = w_1 \dots w_n \in \Sigma^*$ and a computation $P = q_0 \dots q_n$ of M on w we define a step sequence $s_0 := \text{init}(q_0)$ and $s_i := \text{step}(q_{i-1}, w_i, q_i, s_{i-1})$ for $i \in [n]$. We also write $\text{prop}(w, P) := \text{cond}(s_n)$ to be the evaluation of the property.*

These stepwise computation properties now enable us, given a simple NFA, to construct NFAs computing both of the following:

- **Lemma 3.4.** *Let M_f be a simple NFA computing a function f and let prop be a stepwise computation property. Then there is an NFA M computing $g(w) = \sum_P \mathbf{w}_f(P) \cdot \text{prop}(w, P)$, where the sum is over all computations P of M_f on w .*
- **Lemma 3.5.** *Let M_f be a simple NFA computing a function f and let prop be a stepwise computation property. Then there is an NFA M computing $g(w) = \sum_P \text{prop}(w, P)$, where the sum is over all computations P of M_f on w .*

Proof sketch. For both of these lemmas, we construct a sort of product automaton of M_f and prop (represented by the set S), the details can be found in the appendix. ◀

In other words, stepwise computation properties allow us to either “disable” specific computations of M_f or they allow us to directly extract information about the computations of M_f . Note that the restriction on the finiteness of S is necessary, as the elements of S are hard-coded into the state space of the NFA in Lemmas 3.4 and 3.5. In particular, these lemmas do not hold for even countably infinite S , which can be seen with the example $S = \mathbb{N}$ when we define the stepwise computation property in such a way that $\text{prop}(w, P) = 1$ iff $|w|$ is prime, leading to an NFA recognizing the language of all words of prime length, a well known contradiction.

Further note that these two constructions do not incur exponential blowups themselves, however for most of our applications the set S will be of exponential size in the number of states of M_f .

Returning to our decrementation example, to use Lemma 3.4 we want to construct a stepwise computation property prop with $\text{prop}(w, P) = 0$ iff P is the lexicographically smallest non-zero weight computation on w . For this we can set $S = \mathcal{P}(Q)$ to be the set of all subsets of states Q , denoting the set of states that currently are the endpoints of lexicographically smaller partial computations of non-zero weight than the partial computation P we are on. We need to store all such potential states, since some current partial non-zero weight computations might not be possible to be completed to a full non-zero weight computation. Initially this set contains all states that are smaller than the start state of P , the step function then checks which lexicographically smaller partial computations can be extended and whether any new partial computations that agreed with P up to this state can be lexicographically smaller than P . Finally the cond function then checks whether there are any such lexicographically smaller partial computations left that can be completed to a non-zero weight computation, i.e. that end on a state $q \in Q$ with $\text{out}(q) = 1$.

We want to generalize this idea to be able to generally create stepwise computation properties that argue about the number of non-zero computations, either in total or lexicographically smaller than a given computation. However doing this in general would require choosing the set S as the set of functions $Q \rightarrow \mathbb{N}$, which is infinite. As a result we embed the number of computations into finite semirings first to then extract the relevant information. For this purpose we only consider semirings with both additive and multiplicative identities. Homomorphisms h from a semiring $\mathcal{R}$ into another semiring $\mathcal{R}'$ need to fulfill $h(a + b) = h(a) + h(b)$, $h(a \cdot b) = h(a) \cdot h(b)$, $h(1_{\mathcal{R}}) = 1_{\mathcal{R}'}$, $h(0_{\mathcal{R}}) = 0_{\mathcal{R}'}$. Since every element of $\mathbb{N}$ is either 0 or can be formed by repeated addition of 1, any homomorphism from $\mathbb{N}$ into any other semiring is uniquely defined.

We can then use $\mathcal{R}$ to construct stepwise computation properties (the full proofs can be found in the appendix). Combined with Lemmas 3.4 and 3.5 these use similar ideas to [16].

► **Lemma 3.6.** *Let $N = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ be a simple NFA and let $\mathcal{R}$ be a finite semiring and let $\tau : \mathbb{N} \rightarrow \mathcal{R}$ be the unique homomorphism from $\mathbb{N}$ to $\mathcal{R}$. For any function $\pi : \mathcal{R} \rightarrow \{0, 1\}$ there is a stepwise computation property prop with $\text{prop}(w, P) = \pi(\tau(\sum_{P'} \mathbf{w}(P')))$, where the sum is over all computations P' of N on w , independent of P .*

Proof sketch. Construct $\text{prop} = (S, \text{init}, \text{step}, \text{cond})$ via: $S = Q \rightarrow \mathcal{R}$, $\text{init}(q) = r \mapsto \tau(\text{in}(r))$, $\text{step}(q, \sigma, q', s) = r \mapsto \sum_{r' \in Q} s(r') \cdot \tau(\text{wt}(r', \sigma, r))$, and $\text{cond}(s) = \pi(\sum_{r \in Q} s(r) \cdot \tau(\text{out}(r)))$. Let $s_0, \dots, s_n$ be the step sequence of any computation P of w . Since τ is a homomorphism, we can pull out τ . Thus $s_i(q) = \tau(\sum_{\tilde{P}} \mathbf{w}(\tilde{P}))$, where the sum is over all computations $\tilde{P}$ of $w_1, \dots, w_i$ ending in the state q . The condition cond then completes this to $\text{prop}(w, P) = \pi(\tau(\sum_{P'} \mathbf{w}(P')))$, where the sum is over all computations P' of N on w . ◀

► **Lemma 3.7.** *Let $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ be a simple NFA with some ordering $<$ of Q , let $\mathcal{R}$ be a finite semiring and let $\tau : \mathbb{N} \rightarrow \mathcal{R}$ be the unique homomorphism from $\mathbb{N}$ to $\mathcal{R}$. For any function $\pi : \mathcal{R} \rightarrow \{0, 1\}$ there is a stepwise computation property prop with $\text{prop}(w, P) = \pi(\tau(\sum_{P'} \mathbf{w}(P')))$ for all non-zero computations P of N on w , where the sum is over all computations P' of N on w that are lexicographically smaller than P .*

Proof sketch. Construct $\text{prop} = (S, \text{init}, \text{step}, \text{cond})$ via: $S = Q \rightarrow \mathcal{R}$, $\text{init}(q) = r \mapsto \tau(\mathbb{1}_{r < q} \cdot \text{in}(r))$, $\text{step}(q, \sigma, q', s) = r \mapsto \sum_{r' \in Q} s(r') \cdot \tau(\text{wt}(r', \sigma, r)) + \tau(\mathbb{1}_{r < q'} \cdot \text{wt}(q, \sigma, r))$, $\text{cond}(s) = \pi(\sum_{r \in Q} \tau(\text{out}(r)) \cdot s(r))$. Let $s_0, \dots, s_n$ be the step sequence of any computation $P = q_0 \dots q_n$ of w . Inductively we can show that $s_i(r) = \sum_{P'} \tau(\mathbf{w}(P'))$ for all $i \in \{0, \dots, n\}$ and $r \in Q$, where the sum is over all computations $P' = q'_0 \dots q'_i$ of N on $w_1 \dots w_i$ with $q'_i = r$ that are lexicographically smaller than $q_0 \dots q_i$. Initially the only computations $P' = q'_0$ that are lexicographically smaller than the computation q_0 are the ones with $q'_0 < q_0$. For $i > 0$ for a computation $P' = q'_0 \dots q'_i$ to be lexicographically smaller than $q_0 \dots q_i$ there are two possibilities. Either $q'_0 \dots q'_{i-1}$ is already lexicographically smaller than $q_0 \dots q_{i-1}$ or $q'_0 \dots q'_{i-1} = q_0 \dots q_{i-1}$ and $q'_i < q_i$. In the second case the weight of P' is precisely $w(q_{i-1}, w_i, q'_i)$ since P is a computation of non-zero weight and thus weight exactly 1. Combining all of this, we can finish the proof of the claim with a similar argument to Lemma 3.6. ◀

Note that the previous lemma makes no statement about the value of $\text{prop}(w, P)$ for any computations P of weight zero. However, this is enough for our uses, since we only combine it with Lemma 3.4, i.e., $\text{prop}(w, P)$ gets weighted by $\mathbf{w}(P)$.

Most commonly, as is the case for decrementation, we want to be able to exactly distinguish the number of non-zero computations if it is less than k and otherwise be able to tell that the number is at least k . This is achieved by using the following capped semiring:

► **Definition 3.8** (Capped semiring). *For $k \in \mathbb{N}$ we call the semiring $\mathcal{R}_k = \{0, \dots, k\}$ with the operations $a +_{\mathcal{R}} b := \min(a + b, k)$ and $a \cdot_{\mathcal{R}} b := \min(a \cdot b, k)$ the capped semiring.*

We can now show that decrementation is a functional closure property by simply using Lemma 3.7 using the capped semiring $\mathcal{R}_1$ and $\pi(a) = a$ to construct our wanted stepwise computation property computing $\text{prop}(w, P) = 0$ iff P is the lexicographically smallest non-zero weight computation on w .

Most of the remaining closure properties are now proven by using the capped semiring of a specific size and choosing the function π accordingly, we will show this in detail for the example of subtraction, the remaining proofs can be found in the appendix.

► **Lemma 3.9** (Subtraction of constants). *If $f \in \#FA$, then $\forall c \in \mathbb{N} : \max(f - c, 0) \in \#FA$.*

Proof. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$ be a simple NFA computing f with an arbitrary ordering $<$ on Q_f . Lemma 3.7 on the capped semiring $\mathcal{R}_c$ with $\pi(a) = \mathbb{1}_{a \geq c}$ for all $a \in \mathcal{R}$ constructs a stepwise computation property prop with

$$\text{prop}(w, P) = \begin{cases} 1 & \text{if the number of non-zero computations } P' \text{ on } w \text{ that are lex. smaller} \\ & \text{than } P \text{ is at least } c \\ 0 & \text{otherwise} \end{cases}$$

for all computations P on w of non-zero weight, i.e., $\text{prop}(w, P) = 0$ iff P is one of the c lexicographically smallest computations on w with non-zero weight. It follows that the NFA M constructed by Lemma 3.4 computes $g(w) = \max(f(w) - c, 0)$. ◀

If instead of rejecting the c lexicographically smallest computations, we accept only those computations, we compute the minimum of f and c .

► **Lemma 3.10** (Clamping). *If $f \in \#FA$, then $\min(f, c) \in \#FA$ for any constant $c \in \mathbb{N}$.*

By using the capped semiring $\mathcal{R}_{c+1}$ with $\pi_=(a) = \mathbb{1}_{a=c}$, $\pi_\leq(a) = \mathbb{1}_{a \leq c}$ and $\pi_\geq(a) = \mathbb{1}_{a \geq c}$, we can compute the indicator functions $\mathbb{1}_{f=c}$, $\mathbb{1}_{f \leq c}$ and $\mathbb{1}_{f \geq c}$ respectively.

► **Lemma 3.11** (Comparison with constants). *If $f \in \#FA$, then the functions $\mathbb{1}_{f=c}$, $\mathbb{1}_{f \leq c}$, $\mathbb{1}_{f \geq c}$ are in $\#FA$ for any constant $c \in \mathbb{N}$.*

The previous lemma in particular also implies the following:

► **Lemma 3.12.** *If $\varphi : \mathbb{N} \rightarrow \mathbb{N}$ is a functional closure property of $\#FA$ and $\psi : \mathbb{N} \rightarrow \mathbb{N}$ is an arbitrary function with $\varphi(n) = \psi(n)$ for all but finitely many $n \in \mathbb{N}$, then ψ is also a functional closure property of $\#FA$.*

Proof. Let $f \in \#FA$ be arbitrary. Further let $N \in \mathbb{N}$ be such that $\varphi(n) = \psi(n)$ for all $n \geq N$. Then $(\psi \circ f)(w) = \sum_{i=0}^{N-1} \mathbb{1}_{f(w)=i} \psi(i) + \mathbb{1}_{f(w) \geq N} \varphi(f(w))$ for all $w \in \Sigma^*$. In particular $\psi \circ f \in \#FA$ since the $\psi(i)$ are constants and $\varphi \circ f \in \#FA$. ◀

Division and modular arithmetic however use a different semiring: they use the finite cyclic semiring $\mathbb{Z}_c = \{0, \dots, c-1\}$ with $\pi(a) = \mathbb{1}_{a=c-1}$ and $\pi(a) = \mathbb{1}_{a=d}$ respectively.

► **Lemma 3.13** (Division by constants). *If $f \in \#FA$, then $\forall c \in \mathbb{N} \setminus \{0\} : \lfloor f/c \rfloor \in \#FA$.*

► **Lemma 3.14** (Modular arithmetic). *If $f \in \#FA$, then the function $\mathbb{1}_{f \equiv_c d}$ is in $\#FA$ for any constants $c \in \mathbb{N} \setminus \{0\}$ and $d \in \mathbb{Z}_c$.*

The previous closure properties turn out to already be sufficient to generate all functional closure properties, so in particular they are sufficient to generate binomial coefficients by using subtraction of constants, multiplication and division by constants by using the definition of binomial coefficients as a polynomial: $\binom{x}{c} = \frac{1}{c!} x \cdot (x-1) \cdot \dots \cdot (x-c+1)$. Nonetheless, we give an additional proof for binomial coefficients as a different interesting application of the stepwise computation property framework.

► **Lemma 3.15** (Binomial coefficients). *If $f \in \#FA$, then $\binom{f}{c} \in \#FA$ for any constant $c \in \mathbb{N}$.*

Proof. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$ be a simple NFA computing f and let $c \in \mathbb{N}$. For $c < 2$ the statement of this lemma is trivially true, so assume $c \geq 2$.

We construct the c -fold product automaton $M_f^c = (Q_f^c, \Sigma, \text{wt}_f^c, \text{in}_f^c, \text{out}_f^c)$ with

$$\begin{aligned} \text{wt}_f^c((q_1, \dots, q_c), \sigma, (q'_1, \dots, q'_c)) &= \prod_{i=1}^c \text{wt}_f(q_i, \sigma, q'_i) \\ \text{in}_f^c((q_1, \dots, q_c)) &= \prod_{i=1}^c \text{in}_f(q_i) \\ \text{out}_f^c((q_1, \dots, q_c)) &= \prod_{i=1}^c \text{out}_f(q_i) \end{aligned}$$

M_f^c is a simple NFA and every computation on M_f^c is the cartesian product of c computations on M_f . Our aim is to now construct a stepwise computation property $\text{prop} = (S, \text{init}, \text{step}, \text{cond})$ such that $\text{prop}(w, P) = 1$ iff P is composed of c pairwise distinct computations⁸ on N_f .

For this let S be the set of all equivalence relations on the set $[c]$. We define $\text{init}((q_0, \dots, q_c))$ to be the equivalence relation R_0 with $(a, b) \in R_0$ iff $q_a = q_b$. Additionally we define $\text{cond}(R) = 1$ iff R is the equivalence relation where every element is only equivalent to itself, i.e. a computation gets accepted iff all its constituent computations are pairwise distinct. Finally we define $\text{step}((q_1, \dots, q_c), \sigma, (q'_1, \dots, q'_c), R)$ to be the equivalence relation R' defined via $(a, b) \in R'$ iff $(a, b) \in R$ and $q'_a = q'_b$. With a simple induction we can prove that for a computation $P = P_1 \times \dots \times P_c$ and the step sequence $R_0, \dots, R_n$ we have $(a, b) \in R_i$ iff the computations P_a and P_b are identical for the first i steps. It follows that the NFA M constructed by Lemma 3.4 computes $g(w) = \binom{f(w)}{c} \cdot c!$. Now, $\binom{f}{c} \in \#FA$ by Lemma 3.13. ◀

⁸ We could also require them to be sorted in lexicographical order by having S be the set of all total preorders, but since we can divide by $c!$ we are going with the easier exposition.

While the combination of the previous lemmas can be used to show that any polynomial written in the binomial basis with non-negative integer coefficients is a functional closure property of $\#FA$, we can do better by considering a shifted binomial basis. For example, consider the polynomial $\varphi(x) = \frac{x^2}{2} - \frac{3x}{2} + 1$. This polynomial is non-negative for all $x \in \mathbb{N}$. Writing φ in the binomial basis we get $\varphi(x) = \binom{x}{2} - \binom{x}{1} + 1$. If however we allow the upper indices of the binomial basis to be shifted, we can write φ without the use of negative coefficients as $\varphi(x) = \binom{x-1}{2}$. While $x-1$ itself is not a functional closure property of $\#FA$ the function $\max(x-1, 0)$ is a functional closure property of $\#FA$ and is different from $x-1$ for only finitely many $x \in \mathbb{N}$. In the same way we see that $\varphi'(x) := \binom{\max(x-1, 0)}{2}$ only differs from φ for finitely many $x \in \mathbb{N}$, namely $x = 0$. Using Lemma 3.12 to change those finitely many values, we see that φ is indeed a functional closure property of $\#FA$.

Generalizing this idea we will show with the next two lemmas that this is possible for any φ with integer coefficients in the binomial basis, with a small restriction: We don't show that φ itself is a functional closure property of $\#FA$, but rather that $x \mapsto \max(\varphi(x), 0)$ is one. Note that this restriction is the best we can hope for, since no computation in an NFA can ever have negative weight.

► **Lemma 3.16.** *Let $\varphi(x) = \sum_{i=0}^r a_i \cdot \binom{x}{i}$ with $a_i \in \mathbb{Z}$ and $a_r > 0$. Then there are $b_0, \dots, b_r \in \mathbb{N}$ and $c_0, \dots, c_r \in \mathbb{N}$ with $\varphi(x) = \sum_{i=0}^r b_i \cdot \binom{x-c_i}{i}$.*

Proof sketch. We inductively prove this claim by using the Chu-Vandermonde identity [23] on the term of highest degree. It allows us to replace the highest degree binomial via $\binom{x-c_r}{r} = \sum_{i=0}^r (-1)^{r-i} \binom{r-i+c_r-1}{r-i} \binom{x}{i}$. For sufficiently large $c_r \in \mathbb{N}$ this implies that the leading term of $\varphi(x) - a_r \cdot \binom{x-c_r}{r}$ again is positive and of smaller degree. ◀

A full proof of the previous lemma can be found in the appendix.

► **Lemma 3.17 (Integer-valued polynomials).** *Let $f \in \#FA$ and let $\varphi : \mathbb{Q} \rightarrow \mathbb{Q}$ be an integer-valued polynomial, then $\max(\varphi \circ f, 0) \in \#FA$.*

Proof. We can assume the leading coefficient of φ to be positive. Otherwise $\max(\varphi \circ f, 0)$ can be directly written as a finite sum $\sum_i c_i \cdot \mathbb{1}_{f=i}$ which is in $\#FA$ by Lemmas 3.1, 3.2 and 3.11. Write φ in the binomial basis as $\varphi(x) = a_0 \cdot \binom{x}{0} + \dots + a_r \cdot \binom{x}{r}$ with $a_r > 0$. Since φ is integer-valued, all of the a_i are integers, see [14, Prop. 4.2.1]. Using Lemma 3.16 we get a representation $\varphi(x) = \sum_{i=0}^r b_i \cdot \binom{x-c_i}{i}$ with $b_0, \dots, b_r \in \mathbb{N}$ and $c_0, \dots, c_r \in \mathbb{N}$. For $x \geq \max\{c_i \mid 0 \leq i \leq r\} =: N$ we have that $\binom{x-c_i}{i} = \binom{\max(x-c_i, 0)}{i}$ and thus $\psi(x) := \sum_{i=0}^r b_i \cdot \binom{\max(x-c_i, 0)}{i}$ only differs from $x \mapsto \max(\varphi(x), 0)$ on finitely many inputs and is a functional closure property of $\#FA$ by Lemmas 3.1, 3.2, 3.9 and 3.15. Lemma 3.12 then finishes off the claim. ◀

We now have the tools available to show our claim that every ultimately PORC function is a functional closure property of $\#FA$.

► **Lemma 3.18.** *Every ultimately PORC function is a functional closure property of $\#FA$.*

Proof. Let $\varphi : \mathbb{N} \rightarrow \mathbb{N}$ be an ultimately PORC function with period p comprised of the polynomial constituents $\varphi_0, \dots, \varphi_{p-1} : \mathbb{N} \rightarrow \mathbb{Q}$ and $N \in \mathbb{N}$, s.t. for all $n \geq N$ we have $\varphi(n) = \varphi_{n \bmod p}(n)$. Additionally let $f \in \#FA$. We can write

$$\varphi \circ f = \sum_{i=0}^{N-1} \mathbb{1}_{f=i} \cdot \varphi(i) + \mathbb{1}_{f \geq N} \cdot \left(\sum_{i=0}^{p-1} \mathbb{1}_{f \equiv_p i} \cdot \lfloor \max(\varphi_i \circ f, 0) \rfloor \right).$$

Combining Lemmas 3.1, 3.2, 3.11 and 3.14 this shows $\varphi \circ f \in \#FA$, if we can show $\lfloor \max(\varphi_i \circ f, 0) \rfloor \in \#FA$ for all $i \in \{0, \dots, p-1\}$. To show this let α_i be the common denominator of the coefficients of φ_i . Then $\alpha_i \cdot \varphi_i$ is a polynomial with integer coefficients, so it in particular is an integer-valued polynomial and by Lemma 3.17 we have that $\max(\alpha_i \cdot \varphi_i \circ f, 0) \in \#FA$. Combining this with Lemma 3.13 we get that $\left\lfloor \frac{\max(\alpha_i \cdot \varphi_i \circ f, 0)}{\alpha_i} \right\rfloor = \lfloor \max(\varphi_i \circ f, 0) \rfloor \in \#FA$. ◀



■ **Figure 1** NFAs computing the functions $1^n \mapsto n$ and $1^n \mapsto 2^n$ respectively. Edges with a multiplicity of 2 are denoted by listing the edge label twice.

The remainder of this section is dedicated to showing that no other functional closure properties of $\#FA$ exist. This will make use of the following well known algebraic interpretation of NFAs:

► **Lemma 3.19** (see [21]). *If $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ is an NFA, then there are matrices $A_\sigma \in \mathbb{N}^{|Q| \times |Q|}$ for each symbol $\sigma \in \Sigma$ and vectors $a, b \in \mathbb{N}^{|Q|}$, s.t. M computes $a^T \cdot \left(\prod_{j=1}^{|w|} A_{w_j} \right) \cdot b$ for all $w \in \Sigma^*$.*

Proof. We index A_σ , a and b using states $q, q' \in Q$. Choose $(A_\sigma)_{q,q'} = \text{wt}(q, \sigma, q')$, $a_q = \text{in}(q)$ and $b_{q'} = \text{out}(q')$. It is now easy to see that M computes exactly $a^T \cdot \left(\prod_{j=1}^{|w|} A_{w_j} \right) \cdot b$. ◀

When restricting to a unary alphabet $\Sigma = \{\sigma\}$, this degenerates the computed function to $a^T \cdot A_\sigma^{|w|} \cdot b$. In order to analyze the behaviour of these functions we first analyze the behaviour of the matrix power as a function in $|w|$ in the next two lemmas. Their proofs can be found in the appendix.

► **Lemma 3.20.** *Let $A \in \mathbb{N}^{k \times k}$. Then any diagonal entry of A^n is a function $f : \mathbb{N} \rightarrow \mathbb{N}$ with one of the following properties:*

1. $f(n) = 0$ for all $n \in \mathbb{N} \setminus \{0\}$ and $f(0) = 1$.
2. There is a $p \in \mathbb{N} \setminus \{0\}$, such that for all $n \in \mathbb{N}$ we have $f(n) = \mathbb{1}_{n \equiv_p 0}$.
3. There is a $p \in \mathbb{N} \setminus \{0\}$ and a function $g \in 2^{\Theta(n)}$, such that for all $n \in \mathbb{N}$ we have $f(n) = \mathbb{1}_{n \equiv_p 0} \cdot g(n)$.

These naturally correspond to vertices v in the multigraph defined by the adjacency matrix A with

1. no paths from v to v .
2. exactly one path from v to v of length p .
3. multiple walks from v to v where the lengths of all the walks from v to v have gcd p .

We can then lift this result to all entries of A^n .

► **Lemma 3.21.** *If $A \in \mathbb{N}^{k \times k}$, then each entry of A^n is an ultimately almost PORC function.*

► **Theorem 3.22** (Classification of univariate functional closure properties of $\#FA$). *A function $\varphi : \mathbb{N} \rightarrow \mathbb{N}$ is a functional closure property of $\#FA$ iff φ is an ultimately PORC function. This even holds when $\#FA$ is restricted to unary languages.*

Proof. Lemma 3.18 already shows that every ultimately PORC function is a closure property of $\#FA$. It remains to show that all functional closure properties of $\#FA$ are ultimately PORC functions. For this let $\varphi : \mathbb{N} \rightarrow \mathbb{N}$ be a functional closure property of $\#FA$. The function $f : \{1\}^* \rightarrow \mathbb{N}$ defined by $f(1^n) = n$ is computed by the left NFA in Figure 1 and thus in $\#FA$. Consequently $\varphi \circ f \in \#FA$. Let $M = (Q, \{1\}, \text{wt}, \text{in}, \text{out})$ be an NFA computing $\varphi \circ f$. By Lemma 3.19 this NFA induces a transition matrix $A \in \mathbb{N}^{|Q| \times |Q|}$ and vectors $a, b \in \mathbb{N}^{|Q|}$, s.t. $a^T A^n b = \varphi(f(1^n)) = \varphi(n)$ for all $n \in \mathbb{N}$. Every entry of A^n is an ultimately almost PORC function by Lemma 3.21 and thus $a^T A^n b = \varphi(n)$ is one as well. Let p be the quasiperiod of φ and let φ_i be one of the constituents of φ corresponding to some residue class $i \in \{0, \dots, p-1\}$. Assume for the sake of contradiction that φ_i grows in $2^{\Theta(n)}$, i.e. there is a constant $\gamma \in \mathbb{R}^+$ and $N \in \mathbb{N}$, s.t. for every $n \geq N$ we have $\varphi_i(n) \geq 2^{\gamma n}$. Consider the function $f_{p,i} : \{1\}^* \rightarrow \mathbb{N}$

defined by $f_{p,i}(1^n) = p \cdot (2^n + N) + i$. We claim $f_{p,i}$ is in $\#FA$. The function $1^n \mapsto 2^n$ is computed by the right NFA in Figure 1 and thus in $\#FA$. The remainder of the claim follows by Lemma 3.1 and Lemma 3.2. Note that $f_{p,i}(1^n) \equiv_p i$, so $\varphi \circ f_{p,i} = \varphi_i \circ f_{p,i}$ has to be in $\#FA$ as well. Furthermore $\varphi(f_{p,i}(1^n)) = \varphi_i(f_{p,i}(1^n)) = \varphi_i(p \cdot (2^n + N) + i) \geq 2^{\gamma p \cdot (2^n + N) + \gamma i}$ for all $n \in \mathbb{N}$ which is larger than any NFA can compute, since NFAs can only compute functions that are at most linearly exponential in the length of the input. We conclude that none of the constituents φ_i of φ can be exponential, so they are instead all polynomials, making φ an ultimately PORC function. $\blacktriangleleft$

3.2 Multivariate functional closure properties

► **Theorem 3.23** (Classification of multivariate functional closure properties of $\#FA$). *A function $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ is a functional closure property of $\#FA$ iff φ can be written as a finite sum of finite products of univariate ultimately PORC functions.*

Proof. By Theorem 3.22 any univariate ultimately PORC function is a closure property of $\#FA$. As such any finite sum or finite product of them is also a closure property of $\#FA$ by Lemmas 3.1 and 3.2.

It remains to show that all functional closure properties of $\#FA$ are of this form. For this let $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ be a functional closure property of $\#FA$. Define the alphabet $\Sigma = \{\sigma_1, \dots, \sigma_m\}$ and the functions $f_i : \Sigma^* \rightarrow \mathbb{N}$ where $f_i(w) := \#_i(w)$ is defined as the number of occurrences $\#_i(w)$ of the symbol σ_i in w . Applying the closure property to $f_1, \dots, f_m$ gives that $\varphi \circ (f_1, \dots, f_m) \in \#FA$ and thus is computed by an NFA $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$. This induces transition matrices $A_\sigma \in \mathbb{N}^{|Q| \times |Q|}$ for each symbol $\sigma \in \Sigma$ and vectors $a, b \in \mathbb{N}^{|Q|}$, s.t. $a^T \prod_{j=1}^{|w|} A_{w_j} b = \varphi(f_1(w), \dots, f_m(w))$ for all $w \in \Sigma^*$. Restricting to words of the form $w = \sigma_1^{n_1} \sigma_2^{n_2} \dots \sigma_m^{n_m}$ for $n_1, \dots, n_m \in \mathbb{N}$ gives $a^T \left(\prod_{i=1}^m A_{\sigma_i}^{n_i} \right) b = \varphi(n_1, \dots, n_m)$. Using Lemma 3.21 on each of the $A_{\sigma_i}^{n_i}$ we see that every entry of $A_{\sigma_i}^{n_i}$ is an ultimately almost PORC function in n_i . Consequently, every entry of $\prod_{i=1}^m A_{\sigma_i}^{n_i}$ is a finite sum of products of different ultimately almost PORC functions and the same holds for $a^T \left(\prod_{i=1}^m A_{\sigma_i}^{n_i} \right) b = \varphi(n_1, \dots, n_m)$.

We now look at the individual summands of φ and prove that we can rewrite each one as a product of ultimately PORC functions by one-by-one rewriting the exponential constituents. For this let $\varphi^{(1)}(n_1) \dots \varphi^{(m)}(n_m)$ be one of the summands of φ where $\varphi^{(1)}, \dots, \varphi^{(m)}$ are all ultimately almost PORC functions, with periods $p_1, \dots, p_m$, offsets $N_1, \dots, N_m$ and constituents $\varphi_0^{(i)}, \dots, \varphi_{p_i-1}^{(i)}$ for each $i \in [m]$. If none of the constituents are exponential we are done. Otherwise let $\varphi_j^{(i)}$ be one of the exponential constituents, let $\gamma \in \mathbb{R}^+$ and let $N \in \mathbb{N}$, s.t. $\varphi_j^{(i)}(n_i) \geq 2^{\gamma n_i}$ for $n_i \geq N$. We claim we can set $\varphi_j^{(i)}(n_i) = 0$ without changing the product $\varphi^{(1)}(n_1) \dots \varphi^{(m)}(n_m)$ for any $n_1, \dots, n_m \in \mathbb{N}$. Call the resulting functions $\psi^{(i)}$ and $\psi_j^{(i)}$. Assume for the sake of contradiction, that there are some $c_1, \dots, c_m \in \mathbb{N}$ where $\varphi^{(1)}(c_1) \dots \varphi^{(m)}(c_m) \neq \varphi^{(1)}(c_1) \dots \varphi^{(i-1)}(c_{i-1}) \cdot \psi^{(i)}(c_i) \cdot \varphi^{(i+1)}(c_{i+1}) \dots \varphi^{(m)}(c_m)$. This implies that $\varphi^{(1)}(c_1) \dots \varphi^{(i-1)}(c_{i-1}) \cdot \varphi^{(i+1)}(c_{i+1}) \dots \varphi^{(m)}(c_m) \neq 0$ and $c_i \geq N_i$ as we didn't change any other functions except $\varphi^{(i)}$ for $n_i \geq N_i$.

Constructing constant functions $f'_k(w) = c_k$ for $k \neq i$ and the function $f'_i(w) = p_i \cdot (2^{|w|} + \max(N_i, N)) + j$ which are all in $\#FA$. We see that $\varphi \circ (f'_1, \dots, f'_m) \in \#FA$. Note that for any $w \in \Sigma^*$ we have $f'_i(w) \geq \max(N_i, N)$ and $f'_i(w) \equiv_{p_i} j$ and thus $\varphi^{(i)} \circ f'_i = \psi_j^{(i)} \circ f'_i$. Combining all of this we again reach a contradiction to the fact that NFAs can only compute at most linearly exponential functions via

$$\begin{aligned}
\varphi(f'_1(w), \dots, f'_m(w)) &\geq \varphi^{(1)}(c_1) \cdots \varphi^{(i-1)}(c_{i-1}) \cdot \varphi^{(i)}(f'_i(w)) \cdot \varphi^{(i+1)}(c_{i+1}) \cdots \varphi^{(m)}(c_m) \\
&\geq \varphi^{(i)}(f'_i(w)) = \varphi_j^{(i)}(f'_i(w)) = \varphi_j^{(i)}(p_i \cdot (2^{|w|} + \max(N_i, N)) + j) \\
&\geq 2^{\gamma p_i \cdot (2^{|w|} + \max(N_i, N)) + \gamma j}.
\end{aligned}$$

Note that the first inequality holds due to all summands of φ being non-negative. ◀

Deciding whether a function φ has such a representation may not always be directly visible, however if φ is a multivariate polynomial we can be more explicit. Every integer-valued multivariate polynomial has integer coefficients when represented in the binomial basis (see [14, Prop. 4.2.1] for a proof of this fact). We say a term $a \cdot \binom{x_1}{d_1} \cdots \binom{x_m}{d_m}$ dominates another term $a' \cdot \binom{x_1}{d'_1} \cdots \binom{x_m}{d'_m}$ if $d_i \geq d'_i$ for all $i \in [m]$. A term is a dominating term of φ if it has non-zero coefficient and it is not dominated by any other term with non-zero coefficient. We can use a similar approach to Lemma 3.16 to rewrite φ as a positive integer linear combination of products of shifted binomials (details can be found in the appendix).

► **Lemma 3.24.** *Let $\varphi(x_1, \dots, x_m) = \sum_{i=1}^r a_i \cdot \prod_{j=1}^m \binom{x_j}{d_{i,j}}$ with $a_i \in \mathbb{Z}$ and the coefficients of the dominating terms being positive. Then there are $a'_1, \dots, a'_{r'} \in \mathbb{N}$ and $c_1, \dots, c_{r'} \in \mathbb{N}$ with $\varphi(x_1, \dots, x_m) = \sum_{i=1}^{r'} a'_i \cdot \prod_{j=1}^m \binom{x_j - c_i}{d_{i,j}}$.*

We can then use a generalization of Lemma 3.17 and replace subtractions $x_i - c'$ by $\max(x_i - c', 0)$. However special care has to be taken for $x_i < c'$, in which case x_i has to be replaced by the corresponding constants first. This adds the additional condition on φ .

► **Lemma 3.25.** *Let $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ be a multivariate polynomial with rational coefficients, such that whenever ψ is formed from φ by replacing any set of variables – including the empty subset – by constants from $\mathbb{N}$, then all dominating terms of ψ have positive coefficients. Then φ is a functional closure property of #FA.*

► **Lemma 3.26.** *A multivariate polynomial $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ with rational coefficients is a functional closure property of #FA iff for every ψ that can be formed from φ by replacing any subset of variables – including the empty set – by constants from $\mathbb{N}$, then all dominating terms of ψ have positive coefficients.*

Proof sketch. Lemma 3.25 already proves that all multivariate polynomials of this form are functional closure properties of #FA. Now let $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ be a multivariate polynomial with rational coefficient and a functional closure property of #FA. By Theorem 3.23 φ can be written as a finite sum of finite products of ultimately PORC functions. Note that the leading coefficient of each constituent of these ultimately PORC functions is positive. By multivariate polynomial interpolation we can now show that φ is already a finite sum of finite products of these constituents. The dominating terms of φ are then formed by products of the leading coefficients of the constituents and thus are positive. ◀

4 Promise closure properties

► **Definition 4.1.** *Let $S \subseteq \mathbb{N}^m$ and let $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ be a function. We call φ a functional promise closure property of #FA with regard to S if for every $f_1, \dots, f_m \in \#FA$ defined on some shared alphabet Σ there is a function $g \in \#FA$ with $g(w) = \varphi(f_1(w), \dots, f_m(w))$ for every $w \in \Sigma^*$ for which $(f_1(w), \dots, f_m(w)) \in S$.*

We now want to show that if S fulfils some property, namely admitting polynomial cluster sequences (we postpone the definition to Definition 4.5), then for every functional promise closure property with regard to S there is a functional closure property of #FA that agrees with it on all tuples in S . In other words we can interpolate the functional promise

closure property φ on all values of S to obtain a functional closure property φ' for all of $\#FA$. The proof for this follows along the following ideas: First, similar to Theorem 3.23, use the functional closure property φ on the unary counting functions to find an equivalent function φ' that is almost a functional closure property. However after this step some of the constituents of the ultimately almost PORC functions might still be exponential. Theorem 3.23 then proceeded by showing that we can replace all these exponential constituents by the constant zero function, as otherwise we were able to reach a contradiction by constructing functions $f'_1, \dots, f'_m$ and an infinite sequence of inputs $w^{(i)}$, such that $\varphi(f'_1(w^{(i)}), \dots, f'_m(w^{(i)}))$ grows doubly exponential in the length of the inputs. However when dealing with functional promise closure properties we have to be more careful when choosing $f'_1, \dots, f'_m$ and the $w^{(i)}$, because we need $(f'_1(w^{(i)}), \dots, f'_m(w^{(i)})) \in S$ to reach a contradiction. Additionally we don't replace the exponential constituents by the constant zero-function but rather a polynomial that behaves the same for small inputs. For this we need a special variant of univariate polynomial interpolation that yields integer-valued polynomials that are non-negative for all inputs from $\mathbb{N}$:

► **Lemma 4.2.** *Let $c_0, \dots, c_N \in \mathbb{N}$. Then there is an integer-valued polynomial $q : \mathbb{Q} \rightarrow \mathbb{Q}$ with $q(n) = c_n$ for all $n \in \{0, \dots, N\}$ and $q(n') \geq 0$ for all $n' \in \mathbb{N}$.*

To be able to hit all of S consistently we use independent binary encodings, that allow us to hit all of $\mathbb{N}^m$.

► **Lemma 4.3 (Folklore).** *The function $f : \{0, 1\}^* \rightarrow \mathbb{N}$ defined by being the value of $w \in \{0, 1\}^*$ interpreted as a binary number is in $\#FA$. Additionally it is possible to extend the domain of f to any alphabet $\Sigma \supseteq \{0, 1\}$ where the value of f is determined while ignoring any symbols not in $\{0, 1\}$.*

For any $n \in \mathbb{N}$ we denote by $\text{bin}(n)$ the unique binary representation of n without leading zeros. We first show the methodology in detail by proving the univariate case.

► **Theorem 4.4.** *Let $S \subseteq \mathbb{N}$. Then any function $\varphi : \mathbb{N} \rightarrow \mathbb{N}$ is a functional promise closure property of $\#FA$ with regard to S iff there is a functional closure property $\psi : \mathbb{N} \rightarrow \mathbb{N}$ of $\#FA$ with $\varphi|_S = \psi|_S$.*

Proof. If such a ψ exists, we directly see that φ is a functional promise closure property of $\#FA$ with respect to S . Indeed for any $f \in \#FA$ we construct $g = \psi \circ f \in \#FA$ and see $g(w) = \varphi(f(w))$ for every $w \in \Sigma^*$ with $f(w) \in S$.

Now on the other hand let φ be any functional promise closure property of $\#FA$ with regard to S . Again define the alphabet $\Sigma = \{1\}$ and the function $f : \Sigma^* \rightarrow \mathbb{N}$ with $f(1^n) := n$. Applying the closure property to f gives that there is some $g \in \#FA$ with $g(1^n) = \varphi(f(1^n)) = \varphi(n)$ for all $n \in S$. Let $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ be an NFA computing g . This induces a transition matrix $A \in \mathbb{N}^{|Q| \times |Q|}$ and vectors $a, b \in \mathbb{N}^{|Q|}$, s.t. $a^T A^n b = g(1^n)$ for all $n \in \mathbb{N}$ by Lemma 3.19. We now define $\chi(n) := a^T A^n b$ and see $\chi|_S = \varphi|_S$. Using Lemma 3.21 on A^n we see that every entry of A^n is an ultimately almost PORC function in n and as such ψ is also an ultimately almost PORC function. Let p be the quasiperiod of χ and let $\chi_0, \dots, \chi_{p-1}$ be the constituents of χ and let $N \in \mathbb{N}$ be the offset after which χ is defined by the constituents.

We claim that we can now replace every exponential constituent by a polynomial one without changing the value of χ for any $n \in S$. For each $i \in \{0, \dots, p-1\}$ we distinguish two cases, depending on whether $S_i := S \cap (p\mathbb{Z} + i)$ is finite or it is infinite. If S_i is a finite set, we replace χ_i with a polynomial that interpolates the same values as χ_i on S_i . Lemma 4.2 ensures that this polynomial is integer-valued and non-negative for all of $\mathbb{N}$. If S_i is an infinite set, we replace χ_i by the constant zero function. Call the resulting ultimately PORC function ψ with constituents ψ_i . Assume for the sake of contradiction there is an $c \in S$, s.t. $\chi(c) \neq \psi(c)$.

Clearly such a c would have to be at least N . Now let i be, s.t. $c \in S_i$. It must hold that $\chi_i(c) \neq \psi_i(c)$. Hence S_i cannot be a finite set, since χ_i and ψ_i agree on S_i . Therefore S_i must be infinite. Since χ_i is exponential there is a $\gamma \in \mathbb{R}^+$ and $N' \in \mathbb{N}$, s.t. $\chi_i(n) \geq 2^{\gamma n}$ for all $n \geq N'$. Let $f' \in \#FA$ be the function of binary evaluation from Lemma 4.3 over the alphabet $\Sigma' = \{0, 1\}$. Then there is a $g' \in \#FA$ with $g'(\text{bin}(n)) = \varphi(f'(\text{bin}(n))) = \chi(f'(\text{bin}(n)))$ for all $n \in S$. Since S_i is infinite, in particular S_i must contain infinitely many values bigger than $\max(N, N')$. For $n \in S_i$ with $n \geq \max(N, N')$ we now have

$$g'(\text{bin}(n)) = \chi(f'(\text{bin}(n))) = \chi(n) = \chi_i(n) \geq 2^{\gamma n} \geq 2^{\gamma 2^{|\text{bin}(n)|-1}},$$

which is a contradiction to NFAs only being able to only compute functions that are at most linearly exponential in the input length. In conclusion S_i cannot be infinite either and thus c itself cannot exist. $\blacktriangleleft$

► **Definition 4.5.** A set $S \subseteq \mathbb{N}^m$ admits polynomial cluster sequences if for every $N_1, \dots, N_m \in \mathbb{N}$, and $p_1, \dots, p_m \in \mathbb{N}$ the projection $\tau : S \rightarrow \{0, \dots, N_1 + p_1 - 1\} \times \dots \times \{0, \dots, N_m + p_m - 1\}$ defined by $\tau(n_1, \dots, n_m) = (n_1 \text{ srem}_{N_1} p_1, \dots, n_m \text{ srem}_{N_m} p_m)$ has the following property: Any preimage T of a singleton set under τ for every $i \in [m]$ has either bounded i -th coordinate or there is a polynomial $q : \mathbb{N} \rightarrow \mathbb{N}$ and an infinite subset $T' \subseteq T$ with unbounded i -th coordinate⁹ and with $\sum_{j=1}^m n_j \leq q(n_i)$ for all $(n_1, \dots, n_m) \in T'$. We call such an infinite subset a polynomial cluster sequence with regards to dimension i .

We call τ the shifted grid projection of S with respect to offsets $N_1, \dots, N_m$ and quasiperiods $p_1, \dots, p_m$.

Intuitively this definition requires that each dimension is either bounded or can grow reasonably quickly together with the other dimensions, even when restricted to inputs from specific shifted residue classes. For example $\{(n^2, n^3) \mid n \in \mathbb{N}\}$ admits polynomial cluster sequences, while $\{(n, 2^n) \mid n \in \mathbb{N}\}$ does not.

► **Theorem 4.6.** Let $S \subseteq \mathbb{N}^m$ admit polynomial cluster sequences. Then any function $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ is a functional promise closure property of $\#FA$ with regard to S iff there is a functional closure property $\psi : \mathbb{N}^m \rightarrow \mathbb{N}$ of $\#FA$ with $\varphi|_S = \psi|_S$.

The technical proof of the theorem can be found in the appendix. There are multiple natural families for the set S such that S admits polynomial cluster sequences which we describe in the following.

► **Lemma 4.7.** Any finite set $S \subseteq \mathbb{N}^m$ admits polynomial cluster sequences.

Proof. Independent of the offsets and quasiperiods and $i \in [m]$ any subset of S always is finite and thus bounded in every dimension. $\blacktriangleleft$

An affine variety is defined as the zero set of a finite number of multivariate polynomials. A special case of affine varieties are *graph varieties* (also just called *graphs*, see [22, §2.4, Exe. 12]). An affine variety S is a graph variety if there exist a finite number of j -variate polynomials $\mu_1, \dots, \mu_k$ such that $S = \{(s_1, \dots, s_j, \mu_1(s_1, \dots, s_j), \dots, \mu_k(s_1, \dots, s_j)) \mid (s_1, \dots, s_j) \in \mathbb{Q}^j\} \subseteq \mathbb{Q}^{j+k}$. We call $s_1, \dots, s_j$ the *free variables*, and the remaining variables the *dependent variables*. We call S a *monotone graph variety* if $\mu_1, \dots, \mu_k$ are all monotone.

► **Lemma 4.8.** Let $S \subseteq \mathbb{Q}^m = \mathbb{Q}^{j+k}$ be a monotone graph variety. Then the set $S \cap \mathbb{N}^m = \mathbb{N}^{j+k}$ admits polynomial cluster sequences.

⁹ note that this property also follows directly from the polynomial bound on the other coordinates in the i -th coordinate, but we have it as part of the definition for clarity.

The proof of this lemma can be found in the appendix. All in all this combines to the following theorem which characterizes the special case of multivariate polynomial functional promise closure properties, the technical details can be found in the appendix.

► **Theorem 4.9.** *Let $S \subseteq \mathbb{Q}^m = \mathbb{Q}^{j+k}$ be a monotone graph variety and let $I = I(S)$ be its vanishing ideal. A multivariate polynomial $\varphi : S \rightarrow \mathbb{N}$ is a functional promise closure property of $\#FA$ with regard to S if and only if there exists a $\psi \in I$ such that $\varphi + \psi$ is a multivariate functional closure property of $\#FA$.*

5 Conclusion

We characterized the functional closure properties of $\#FA$ to be precisely the ultimately PORC functions in the univariate case and combinations of ultimately PORC functions in the multivariate case. Additionally we characterize promise functional closure properties of $\#FA$ with regard to some natural families of sets S . Natural further directions of research are now whether we can characterize the promise functional closure properties of $\#FA$ for more sets S and whether our methods can be applied to characterize functional closure properties for more powerful models of computation.

References

- 1 Rudolf Ahlswede and David E Daykin. An inequality for the weights of two families of sets, their unions and intersections. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 43:183–185, 1978.
- 2 Noga Alon and Joel H Spencer. *The probabilistic method*. John Wiley & Sons, 2016.
- 3 Matthias Beck and Sinai Robins. *Computing the continuous discretely*, volume 61. Springer, 2007.
- 4 Edwin F Beckenbach and Richard Bellman. *Inequalities*. Springer, Berlin, 1961.
- 5 Richard Beigel. Closure properties of GapP and $\#P$. In *Proceedings of the Fifth Israeli Symposium on Theory of Computing and Systems*, pages 144–146. IEEE, 1997.
- 6 Swee Hong Chan and Igor Pak. Computational complexity of counting coincidences. 2023.
- 7 Swee Hong Chan and Igor Pak. Equality cases of the Alexandrov–Fenchel inequality are not in the polynomial hierarchy. 2023.
- 8 David Cox, John Little, and Donal O’Shea. *Ideals, varieties, and algorithms: an introduction to computational algebraic geometry and commutative algebra*. Springer Science & Business Media, 2013.
- 9 Manfred Droste, Werner Kuich, and Heiko Vogler. *Handbook of weighted automata*. Springer Science & Business Media, 2009.
- 10 Manfred Droste and Dietrich Kuske. Weighted automata. pages 113–150. in *Handbook of Automata Theory*, European Mathematical Society.
- 11 Mariano Gasca and Thomas Sauer. Polynomial interpolation in several variables. *Advances in Computational Mathematics*, 12:377–410, 2000.
- 12 HG Hardy, JE Littlewood, and G. Pólya. *Inequalities*. Cambridge University Press, 1952.
- 13 Ulrich Hertrampf, Heribert Vollmer, and Klaus W Wagner. On the power of number-theoretic operations with respect to counting. In *Proceedings of Structure in Complexity Theory. Tenth Annual IEEE Conference*, pages 299–314. IEEE, 1995.
- 14 Christian Ikenmeyer and Igor Pak. What is in $\#P$ and what is not? In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 860–871. IEEE, 2022.
- 15 Christian Ikenmeyer, Igor Pak, and Greta Panova. Positivity of the symmetric group characters is as hard as the polynomial time hierarchy. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3573–3586. SIAM, 2023.

- 16 Ines Klimann, Sylvain Lombardy, Jean Mairesse, and Christophe Prieur. Deciding unambiguity and sequentiality from a finitely ambiguous max-plus automaton. *Theoretical Computer Science*, 327(3):349–373, 2004.
- 17 Igor Pak. Complexity problems in enumerative combinatorics. In *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*, pages 3153–3180. World Scientific, 2018.
- 18 Igor Pak. Combinatorial inequalities. *Notices of the AMS*, 66(7), August 2019.
- 19 Igor Pak. What is a combinatorial interpretation? to appear: Proc. Open Problems in Algebraic Combinatorics. <https://www.samuelhofkins.com/OPAC/files/proceedings/pak.pdf>, 2022.
- 20 J Peterson. Beviser for wilsons og fermats theoremer. *Tidsskrift for matematik*, 2:64–65, 1872.
- 21 Marcel Paul Schützenberger. On the definition of a family of automata. *Inf. Control.*, 4(2-3):245–270, 1961.
- 22 Igor R Shafarevich. *Basic algebraic geometry 1: Varieties in projective space*. Springer Science & Business Media, 3 edition, 2013.
- 23 Michael Spivey. The Chu-Vandermonde identity via Leibniz’s identity for derivatives. *The College Mathematics Journal*, 47(3):219–220, 2016.
- 24 Richard P Stanley. Positivity problems and conjectures in algebraic combinatorics. *Mathematics: frontiers and perspectives*, 295:319, 1999.

A

 Appendix: Missing proofs

A.1 Missing proofs from section 2

Proof of Lemma 2.3. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$ be an NFA computing f . We construct $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ with

$$\begin{aligned} Q &= Q_f \times [\max(\text{im } \text{wt}_f \cup \text{im } \text{in}_f)] \times [\max(\text{im } \text{out}_f)] \\ \text{wt}((q, \alpha, \beta), \sigma, (q', \alpha', \beta')) &= \begin{cases} 1 & \text{if } \alpha' \leq \text{wt}_f(q, \sigma, q') \text{ and } \beta = 1 \\ 0 & \text{otherwise} \end{cases} \\ \text{in}((q, \alpha, \beta)) &= \begin{cases} 1 & \text{if } \alpha \leq \text{in}_f(q) \\ 0 & \text{otherwise} \end{cases} \\ \text{out}((q, \alpha, \beta)) &= \begin{cases} 1 & \text{if } \beta \leq \text{out}_f(q) \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

Intuitively M consists of multiple identical copies of M_f , where all the weights are handled explicitly as separate computations.

Fix any word $w = w_1 \dots w_n \in \Sigma^*$. Let $P = q_0 q_1 \dots q_n$ be a computation of M_f on w . Then for $\alpha_0 \in [\text{in}_f(q_0)]$ and $\alpha_i \in [\text{wt}_f(q_{i-1}, w_i, q_i)]$ for $i \in [n]$ and $\beta_0 = \beta_1 = \dots = \beta_{n-1} = 1$ and $\beta_n \in \text{out}_f(q_n)$ we have that $P' := (q_0, \alpha_0, \beta_0)(q_1, \alpha_1, \beta_1) \dots (q_n, \alpha_n, \beta_n)$ is a computation of M on w of weight exactly 1 while any other choices of the α_i and β_i would lead to a computation of weight 0. In particular there are $\text{in}_f(q_0) \cdot \prod_{i=1}^n \text{wt}_f(q_{i-1}, w_i, q_i) \cdot \text{out}_f(q_n)$ non-zero computations in M corresponding to P , which coincides with the weight of P in M_f . Additionally every computation on M can be projected onto a computation on M_f via $(q, i, j) \mapsto q$ and thus M computes f . ◀

A.2 Missing proofs from section 3

► **Lemma 3.1** (Addition). *If $f, g \in \#FA$, then $f + g \in \#FA$.*

Proof. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$ and $M_g = (Q_g, \Sigma, \text{wt}_g, \text{in}_g, \text{out}_g)$ be NFAs computing f and g respectively.

We construct the direct union NFA $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ with

$$\begin{aligned} Q &= Q_f \dot{\cup} Q_g \\ \text{wt}(q, \sigma, q') &= \begin{cases} \text{wt}_f(q, \sigma, q') & q, q' \in Q_f \\ \text{wt}_g(q, \sigma, q') & q, q' \in Q_g \\ 0 & \text{otherwise} \end{cases} \\ \text{in}(q) &= \begin{cases} \text{in}_f(q) & \text{if } q \in Q_f \\ \text{in}_g(q) & \text{if } q \in Q_g \end{cases} \\ \text{out}(q) &= \begin{cases} \text{out}_f(q) & \text{if } q \in Q_f \\ \text{out}_g(q) & \text{if } q \in Q_g \end{cases} \end{aligned}$$

Any computation with non-zero weight in M is now fully contained in exactly one of the sub-NFA equivalent to M_f or M_g and any computation path on M_f or M_g naturally embeds into M and thus M computes $f + g$. ◀

► **Lemma 3.2** (Multiplication). *If $f, g \in \#FA$, then $f \cdot g \in \#FA$.*

Proof. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$ and $M_g = (Q_g, \Sigma, \text{wt}_g, \text{in}_g, \text{out}_g)$ be NFAs computing f and g respectively. We use the product automaton construction to construct the NFA $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$:

$$\begin{aligned} Q &= Q_f \times Q_g \\ \text{wt}((q_1, q_2), \sigma, (q'_1, q'_2)) &= \text{wt}_f(q_1, \sigma, q'_1) \cdot \text{wt}_g(q_2, \sigma, q'_2) \\ \text{in}((q_1, q_2)) &= \text{in}_f(q_1) \cdot \text{in}_g(q_2) \\ \text{out}((q_1, q_2)) &= \text{out}_f(q_1) \cdot \text{out}_g(q_2) \end{aligned}$$

Fix some $w = w_1 \dots w_n \in \Sigma^*$. Let $\mathcal{P}_f = Q_f^{n+1}$ be all computations of M_f on w and let $\mathcal{P}_g = Q_g^{n+1}$ be all computations of M_g on w . Then $\mathcal{P}_f \times \mathcal{P}_g \cong (Q_f \times Q_g)^{n+1}$ are precisely the computations of M on x by identifying $((q_{f,0} \dots q_{f,n}), (q_{g,0} \dots q_{g,n}))$ with the computation $(q_{f,0}, q_{g,0}) \dots (q_{f,n}, q_{g,n})$. Additionally the computation $(P_f, P_g) \in \mathcal{P}_f \times \mathcal{P}_g$ has weight $\mathbf{w}((P_f, P_g)) = \mathbf{w}_f(P_f) \cdot \mathbf{w}_g(P_g)$ where $\mathbf{w}_f$ and $\mathbf{w}_g$ denote the weights of computations of M_f and M_g respectively. Thus we have

$$\left(\sum_{P_f \in \mathcal{P}_f} \mathbf{w}_f(P_f) \right) \cdot \left(\sum_{P_g \in \mathcal{P}_g} \mathbf{w}_g(P_g) \right) = \sum_{P_f \in \mathcal{P}_f} \sum_{P_g \in \mathcal{P}_g} \mathbf{w}_f(P_f) \cdot \mathbf{w}_g(P_g) = \sum_{(P_f, P_g) \in \mathcal{P}_f \times \mathcal{P}_g} \mathbf{w}_f((P_f, P_g))$$

and M computes $f \cdot g$. ◀

► **Lemma 3.4.** *Let M_f be a simple NFA computing a function f and let prop be a stepwise computation property. Then there is an NFA M computing $g(w) = \sum_P \mathbf{w}_f(P) \cdot \text{prop}(w, P)$, where the sum is over all computations P of M_f on w .*

Proof. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$. We construct the NFA $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ with $Q = Q_f \times S$ and

$$\begin{aligned} \text{in}((q, s)) &= \begin{cases} \text{in}_f(q) & \text{if } s = \text{init}(q) \\ 0 & \text{otherwise} \end{cases} \\ \text{out}((q, s)) &= \text{out}_f(q) \cdot \text{cond}(s) \\ \text{wt}((q, s), \sigma, (q', s')) &= \begin{cases} \text{wt}_f(q, \sigma, q') & \text{if } s' = \text{step}(q, \sigma, q', s) \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

Fix some $w = w_1 \dots w_n \in \Sigma^*$. Any computation $P = (q_0, s_0) \dots (q_n, s_n)$ of M on w directly corresponds to the computation $q_0 \dots q_n$ of M_f . If $s_0 \neq \text{init}(q_0)$ or if $s_i \neq \text{step}(q_{i-1}, w_i, q_i, s_{i-1})$ for any $i \in [n]$ then P has weight 0. Otherwise P has weight $\mathbf{w}(P) = \mathbf{w}_f(q_0 \dots q_n) \cdot \text{cond}(s_n) = \mathbf{w}_f(q_0 \dots q_n) \cdot \text{prop}(w, q_0 \dots q_n)$. The claim on the function computed by M directly follows. ◀

► **Lemma 3.5.** *Let M_f be a simple NFA computing a function f and let prop be a stepwise computation property. Then there is an NFA M computing $g(w) = \sum_P \text{prop}(w, P)$, where the sum is over all computations P of M_f on w .*

Proof. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$. We construct the NFA $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ with $Q = Q_f \times S$ and

$$\begin{aligned} \text{in}((q, s)) &= \begin{cases} 1 & \text{if } s = \text{init}(q) \\ 0 & \text{otherwise} \end{cases} \\ \text{out}((q, s)) &= \text{cond}(s) \\ \text{wt}((q, s), \sigma, (q', s')) &= \begin{cases} 1 & \text{if } s' = \text{step}(q, \sigma, q', s) \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

Any computation $P = (q_0, s_0) \dots (q_n, s_n)$ of M directly corresponds to the computation $q_0 \dots q_n$ of M_f . If $s_0 \neq \text{init}(q_0)$ or if $s_i \neq \text{step}(q_{i-1}, w_i, q_i, s_{i-1})$ for any $i \in [n]$ then P has weight 0. Otherwise P has weight $\mathbf{w}(P) = \text{cond}(s_n) = \text{prop}(w, q_0 \dots q_n)$. The claim on the function computed by M directly follows. $\blacktriangleleft$

► **Lemma 3.6.** *Let $N = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ be a simple NFA and let $\mathcal{R}$ be a finite semiring and let $\tau : \mathbb{N} \rightarrow \mathcal{R}$ be the unique homomorphism from $\mathbb{N}$ to $\mathcal{R}$. For any function $\pi : \mathcal{R} \rightarrow \{0, 1\}$ there is a stepwise computation property prop with $\text{prop}(w, P) = \pi(\tau(\sum_{P'} \mathbf{w}(P')))$, where the sum is over all computations P' of N on w , independent of P .*

Proof. We construct $\text{prop} = (S, \text{init}, \text{step}, \text{cond})$ as follows:

$$\begin{aligned} S &= Q \rightarrow \mathcal{R} \\ \text{init}(q) &= r \mapsto \tau(\text{in}(r)) \\ \text{step}(q, \sigma, q', s) &= r \mapsto \sum_{r' \in Q} s(r') \cdot \tau(\text{wt}(r', \sigma, r)) \\ \text{cond}(s) &= \pi\left(\sum_{r \in Q} s(r) \cdot \tau(\text{out}(r))\right) \end{aligned}$$

Note that step and init depend on neither q nor q' . So for a fixed $w = w_1 \dots w_n \in \Sigma^*$ the step sequence $s_0, \dots, s_n$ collapses to $s_0 = r \mapsto \tau(\text{in}(r))$ and $s_i = r \mapsto \sum_{r' \in Q} s_{i-1}(r') \cdot \tau(\text{wt}(r', w_i, r))$ for $i \in [n]$ independent of the specific computation of N on x .

We first prove that for all $w = w_1 \dots w_n \in \Sigma^*$ and $q \in Q$ we have $s_n(r) = \tau(\sum_{P_r} \mathbf{w}(P_r))$ where the sum is over all computations $P_r = q_0 \dots q_n$ of N on w with $q_n = r$. We prove this by induction over n . For $n = 0$ this holds directly. For $n > 0$ we have

$$\begin{aligned} s_n(r) &= \sum_{q_{n-1} \in Q} s_{n-1}(q_{n-1}) \cdot \tau(\text{wt}(q_{n-1}, w_n, r)) \\ &= \sum_{q_{n-1} \in Q} \tau\left(\sum_{q_0, \dots, q_{n-2} \in Q} \mathbf{w}(q_0 \dots q_{n-1})\right) \cdot \tau(\text{wt}(q_{n-1}, w_n, r)) \\ &= \tau\left(\sum_{q_{n-1} \in Q} \sum_{q_0, \dots, q_{n-2} \in Q} \mathbf{w}(q_0 \dots q_{n-1}) \cdot \text{wt}(q_{n-1}, w_n, r)\right) \\ &= \tau\left(\sum_{q_0, \dots, q_{n-1} \in Q} \mathbf{w}(q_0 \dots q_{n-1} r)\right). \end{aligned}$$

We can thus finish the proof with

$$\begin{aligned}
\text{prop}(w, P) &= \text{cond}(s_n) \\
&= \pi \left(\sum_{q_n \in Q} s_n(r) \cdot \tau(\text{out}(q_n)) \right) \\
&= \pi \left(\sum_{q_n \in Q} \tau \left(\sum_{q_0, \dots, q_{n-1} \in Q} \mathbf{w}(q_0 \dots q_n) \right) \cdot \tau(\text{out}(q_n)) \right) \\
&= \pi \left(\tau \left(\sum_{q_n \in Q} \sum_{q_0, \dots, q_{n-1} \in Q} \mathbf{w}(q_0 \dots q_n) \cdot \text{out}(q_n) \right) \right) \\
&= \pi \left(\tau \left(\sum_{q_0, \dots, q_n \in Q} \mathbf{w}(q_0 \dots q_n) \right) \right). \quad \blacktriangleleft
\end{aligned}$$

► **Lemma 3.7.** *Let $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ be a simple NFA with some ordering $<$ of Q , let $\mathcal{R}$ be a finite semiring and let $\tau : \mathbb{N} \rightarrow \mathcal{R}$ be the unique homomorphism from $\mathbb{N}$ to $\mathcal{R}$. For any function $\pi : \mathcal{R} \rightarrow \{0, 1\}$ there is a stepwise computation property prop with $\text{prop}(w, P) = \pi(\tau(\sum_{P'} \mathbf{w}(P')))$ for all non-zero computations P of N on w , where the sum is over all computations P' of N on w that are lexicographically smaller than P .*

Proof. We construct $\text{prop} = (S, \text{init}, \text{step}, \text{cond})$ as follows:

$$\begin{aligned}
S &= Q \rightarrow \mathcal{R} \\
\text{init}(q) &= r \mapsto \tau(\mathbb{1}_{r < q} \cdot \text{in}(r)) \\
\text{step}(q, \sigma, q', s) &= r \mapsto \sum_{r' \in Q} s(r') \cdot \tau(\text{wt}(r', \sigma, r)) + \tau(\mathbb{1}_{r < q'} \cdot \text{wt}(q, \sigma, r)) \\
\text{cond}(s) &= \pi(\sum_{r \in Q} \tau(\text{out}(r)) \cdot s(r))
\end{aligned}$$

Fix some $w = w_1 \dots w_n \in \Sigma^*$ and some computation $P = q_0 \dots q_n$ of N on w of non-zero weight. This fixes the step sequence $s_0, \dots, s_n$. We now prove by induction over i that $s_i(r) = \sum_{P'} \tau(\mathbf{w}(P'))$ for all $i \in \{0, \dots, n\}$ and $r \in Q$, where the sum is over all computations $P' = q'_0 \dots q'_i$ of $w_1 \dots w_i$ with $q'_i = r$ that are lexicographically smaller than $q_0 \dots q_i$. For $i = 0$ this is trivially the case since the only computations $P' = q'_0$ that are lexicographically smaller than the computation q_0 are the ones with $q'_0 < q_0$. For $i > 0$ for a computation $P' = q'_0 \dots q'_i$ to be lexicographically smaller than $q_0 \dots q_i$ there are two possibilities. Either $q'_0 \dots q'_{i-1}$ is already lexicographically smaller than $q_0 \dots q_{i-1}$ or $q'_0 \dots q'_{i-1} = q_0 \dots q_{i-1}$ and $q'_i < q_i$. In the second case the weight of P' is precisely $\text{wt}(q_{i-1}, w_i, q'_i)$ since P is a computation of non-zero weight and thus weight exactly 1. Combining all of this we can finish the proof of the claim with a similar argument to Lemma 3.6:

$$\begin{aligned}
s_i(r) &= \sum_{q'_{i-1} \in Q} s_{i-1}(q'_{i-1}) \cdot \tau(\text{wt}(q'_{i-1}, w_i, r)) + \tau(\mathbb{1}_{r < q_i} \cdot \text{wt}(q_{i-1}, \sigma, r)) \\
&= \tau \left(\sum_{\substack{q'_0 \dots q'_{i-1} \\ \text{lex. smaller than} \\ q_0 \dots q_{i-1}}} \mathbf{w}(q'_0 \dots q'_{i-1}) \cdot \text{wt}(q'_{i-1}, w_i, r) \right) + \tau(\mathbb{1}_{r < q_i} \cdot \mathbf{w}(q_0 \dots q_{i-1} r)) \\
&= \tau \left(\sum_{\substack{q'_0 \dots q'_{i-1} \\ \text{lex. smaller than} \\ q_0 \dots q_{i-1}}} \mathbf{w}(q'_0 \dots q'_{i-1} r) + \mathbb{1}_{r < q_i} \cdot \mathbf{w}(q_0 \dots q_{i-1} r) \right) \\
&= \tau \left(\sum_{\substack{q'_0 \dots q'_{i-1} r \\ \text{lex. smaller than} \\ q_0 \dots q_{i-1} q_i}} \mathbf{w}(q'_0 \dots q'_{i-1} r) \right)
\end{aligned}$$

The statement of this lemma then follows analogously to Lemma 3.6. $\blacktriangleleft$

► **Lemma 3.10** (Clamping). *If $f \in \#FA$, then $\min(f, c) \in \#FA$ for any constant $c \in \mathbb{N}$.*

Proof. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$ be a simple NFA computing f with an arbitrary ordering $<$ on Q_f . Lemma 3.7 on the capped semiring $\mathcal{R}_c$ with $\pi(a) = \mathbb{1}_{a < c}$ for all $a \in \mathcal{R}$ constructs a stepwise computation property prop with

$$\text{prop}(w, P) = \begin{cases} 1 & \text{if the number of non-zero computations } P' \text{ on } w \text{ that are lex. smaller} \\ & \text{than } P \text{ is at most } c - 1 \\ 0 & \text{otherwise} \end{cases}$$

for all computations P of M_f on w of non-zero weight, i.e. $\text{prop}(w, P) = 1$ iff P is one of the c lexicographically smallest computations on w with non-zero weight. It follows that the NFA M constructed by Lemma 3.4 computes $g(w) = \min(f(w), c)$. $\blacktriangleleft$

► **Lemma 3.11** (Comparison with constants). *If $f \in \#FA$, then the functions $\mathbb{1}_{f=c}$, $\mathbb{1}_{f \leq c}$, $\mathbb{1}_{f \geq c}$ are in $\#FA$ for any constant $c \in \mathbb{N}$.*

Proof. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$ be a simple NFA computing f with an arbitrary ordering $<$ on Q_f . Lemma 3.6 on the capped semiring $\mathcal{R}_{c+1}$ with $\pi_=(a) = \mathbb{1}_{a=c}$, $\pi_<(a) = \mathbb{1}_{a \leq c}$ and $\pi_>(a) = \mathbb{1}_{a \geq c}$ for all $a \in \mathcal{R}$ constructs stepwise computation properties $\text{prop}_=$, $\text{prop}_\leq$ and $\text{prop}_\geq$ respectively with

$$\begin{aligned} \text{prop}_=(w, P) &= \begin{cases} 1 & \text{if the number of non-zero computations } P' \text{ on } w \text{ is exactly } c \\ 0 & \text{otherwise} \end{cases} \\ \text{prop}_\leq(w, P) &= \begin{cases} 1 & \text{if the number of non-zero computations } P' \text{ on } w \text{ is at most } c \\ 0 & \text{otherwise} \end{cases} \\ \text{prop}_\geq(w, P) &= \begin{cases} 1 & \text{if the number of non-zero computations } P' \text{ on } w \text{ is at least } c \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

for all computations P of M_f on w . It follows that the NFAs $M_=$, $M_\leq$ and $M_\geq$ constructed by Lemma 3.5 compute $g_=(w) = \mathbb{1}_{f(w)=c} \cdot |Q_f|^{|w|+1}$, $g_\leq(w) = \mathbb{1}_{f(w) \leq c} \cdot |Q_f|^{|w|+1}$ and $g_\geq(w) = \mathbb{1}_{f(w) \geq c} \cdot |Q_f|^{|w|+1}$ respectively. Since $|Q_f|^{|w|+1}$ is always positive Lemma 3.10 finishes the proof with

$$\min(g_=(w), 1) = \mathbb{1}_{f(w)=c} \quad \min(g_\leq(w), 1) = \mathbb{1}_{f(w) \leq c} \quad \min(g_\geq(w), 1) = \mathbb{1}_{f(w) \geq c}. \quad \blacktriangleleft$$

► **Lemma 3.13** (Division by constants). *If $f \in \#FA$, then $\forall c \in \mathbb{N} \setminus \{0\} : \lfloor f/c \rfloor \in \#FA$.*

Proof. For $c = 1$ the statement is trivially true, so let $c \geq 2$. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$ be a simple NFA computing f with an arbitrary ordering $<$ on Q_f . We use the finite cyclic semiring $\mathcal{R} = \mathbb{Z}_c$ with the usual addition and multiplication. Lemma 3.7 with $\pi(a) = \mathbb{1}_{a=c-1}$ for all $a \in \mathcal{R}$ constructs a stepwise computation property prop with

$$\text{prop}(w, P) = \begin{cases} 1 & \text{if the number of non-zero computations } P' \text{ on } w \text{ that are} \\ & \text{lex. smaller than } P \text{ is one less than a multiple of } c \\ 0 & \text{otherwise} \end{cases}$$

for all computations P of M_f on w of non-zero weight. It follows that the NFA M constructed by Lemma 3.4 computes $g(w) = \left\lfloor \frac{f(w)}{c} \right\rfloor$. $\blacktriangleleft$

► **Lemma 3.14** (Modular arithmetic). *If $f \in \#FA$, then the function $\mathbb{1}_{f \equiv_c d}$ is in $\#FA$ for any constants $c \in \mathbb{N} \setminus \{0\}$ and $d \in \mathbb{Z}_c$.*

Proof. For $c = 1$ the statement is trivially true, so let $c \geq 2$. Let $M_f = (Q_f, \Sigma, \text{wt}_f, \text{in}_f, \text{out}_f)$ be a simple NFA computing f with an arbitrary ordering $<$ on Q_f . We use the finite cyclic semiring $\mathcal{R} = \mathbb{Z}_c$ with the usual addition and multiplication. Lemma 3.6 with $\pi(a) = \mathbb{1}_{a=d}$ for all $a \in \mathcal{R}$ constructs a stepwise computation property prop with

$$\text{prop}(w, P) = \begin{cases} 1 & \text{if the number of non-zero computations } P' \text{ on } w \\ & \text{is } d \text{ more than a multiple of } c \\ 0 & \text{otherwise} \end{cases}$$

for all computations P of M_f on w of non-zero weight. It follows that the NFA M constructed by Lemma 3.5 computes $g(w) = \mathbb{1}_{f(w) \equiv_c d} \cdot |Q_f|^{|w|+1}$. Since $|Q_f|^{|w|+1}$ is always positive Lemma 3.10 finishes the proof with $\min(g(w), 1) = \mathbb{1}_{f(w) \equiv_c d}$. ◀

► **Lemma 3.16.** *Let $\varphi(x) = \sum_{i=0}^r a_i \cdot \binom{x}{i}$ with $a_i \in \mathbb{Z}$ and $a_r > 0$. Then there are $b_0, \dots, b_r \in \mathbb{N}$ and $c_0, \dots, c_r \in \mathbb{N}$ with $\varphi(x) = \sum_{i=0}^r b_i \cdot \binom{x-c_i}{i}$.*

Proof. We prove this via induction on r . For $r = 0$ we can choose $b_0 = a_0$ and $c_0 = 0$ to see that φ is of the required form. For $r > 0$ we choose $b_r = a_r$ and $c_r = \max\{-a_{r-1} + 1, 0\}$.

We define the polynomial $\varphi'(x) := \varphi(x) - b_r \cdot \binom{x-c_r}{r}$ and analyze the leading binomial $\binom{x-c_r}{r}$. Using the Chu-Vandermonde identity [23] we see $\binom{x-c_r}{r} = \sum_{i=0}^r \binom{-c_r}{r-i} \binom{x}{i} = \sum_{i=0}^r (-1)^{r-i} \binom{r-i+c_r-1}{r-i} \binom{x}{i}$.

Combining this with $\varphi(x) = \sum_{i=0}^r a_i \cdot \binom{x}{i}$ we see that $\varphi(x) - b_r \cdot \binom{x-c_r}{r} = \sum_{i=0}^{r-1} a'_i \cdot \binom{x}{i}$ with $a'_i = a_i - b_r \cdot (-1)^{r-i} \cdot \binom{r-i+c_r-1}{r-i}$. Furthermore we have $a'_{r-1} > 0$ due to $b_r \cdot (-1)^{r-(r-1)} \binom{r-(r-1)+c_r-1}{r-(r-1)} = -b_r c_r < a_{r-1}$. By induction there are $b'_0, \dots, b'_{r-1} \in \mathbb{N}$ and $c'_0, \dots, c'_{r-1} \in \mathbb{N}$ s.t. $\varphi'(x) = \sum_{i=0}^{r-1} b'_i \cdot \binom{x-c'_i}{i}$. Thus by choosing $c_i = c'_i$ and $b_i = b'_i$ for $i \in \{0, \dots, r-1\}$ we get $\varphi(x) = \sum_{i=0}^r b_i \cdot \binom{x-c_i}{i}$. ◀

► **Lemma 3.20.** *Let $A \in \mathbb{N}^{k \times k}$. Then any diagonal entry of A^n is a function $f : \mathbb{N} \rightarrow \mathbb{N}$ with one of the following properties:*

1. $f(n) = 0$ for all $n \in \mathbb{N} \setminus \{0\}$ and $f(0) = 1$.
2. There is a $p \in \mathbb{N} \setminus \{0\}$, such that for all $n \in \mathbb{N}$ we have $f(n) = \mathbb{1}_{n \equiv_p 0}$.
3. There is a $p \in \mathbb{N} \setminus \{0\}$ and a function $g \in 2^{\Theta(n)}$, such that for all $n \in \mathbb{N}$ we have $f(n) = \mathbb{1}_{n \equiv_p 0} \cdot g(n)$.

Proof. To prove the lemma, let v be arbitrary and let $f_v(n) = (A^n)_{vv}$. Note that we always have $f_v(0) = 1$ since A^0 is the identity matrix. We split into three cases:

$f_v(n) = 0$ for all $n > 0$: In this case we are done.

$f_v(n) \in \{0, 1\}$ for all $n \in \mathbb{N}$ and there is some $\ell > 0$ with $f_v(\ell) = 1$: Let $p \geq 1$ be minimal with $f_v(p) = 1$. Then we directly know that for any multiple $i \cdot p$ we have $f_v(ip) = (A^{ip})_{vv} \geq ((A^p)_{vv})^i = f_v(p)^i = 1$. In the setting if A is interpreted as the adjacency matrix of a multigraph, this corresponds to a walk consisting of repeating the same path of length p from v to itself i times.

Assume for the sake of contradiction there is any other n that is not a multiple of p with $f_v(n) = 1$. Then we have that $f_v(n \cdot p) = (A^{np})_{vv} \geq (A^p)_{vv}^n + (A^n)_{vv}^p = f_v(p)^n + f_v(n)^p = 2$, a contradiction. Going back to the graph setting, this corresponds to the choice of either repeating a walk from v to v of length n a total of p times or repeating a walk from v to v of length p a total of n times, leading to two different walks of length $n \cdot p$ from v to v since neither does p divide n nor vice-versa.

There is some $\ell \in \mathbb{N}$ with $f_v(\ell) \geq 2$: Let p be the gcd of the set $Z = \{n \in \mathbb{N} \mid f_v(n) \geq 1\}$.

If $n \in \mathbb{N}$ is not divisible by p we directly have $f_v(n) = 0$ since this implies $n \notin Z$.

Additionally there is some finite subset $Z' \subseteq Z$ with the same gcd p . By Bézout's lemma there is some $N \in \mathbb{N}$ which is a multiple of p s.t. that every number of the form $N + i \cdot p$ with $i \in \mathbb{N}$ can be written as a non-negative linear combination of elements from Z' . Let $n = N + i \cdot p$ for any $i \in \mathbb{N}$. We can write

$$n = N + \underbrace{\frac{n - N - \lfloor \frac{n-N}{\ell} \rfloor \cdot \ell}{p}}_{\alpha:=} \cdot p + \underbrace{\left\lfloor \frac{n - N}{\ell} \right\rfloor}_{\beta:=} \cdot \ell. \quad (1)$$

since $\ell \in Z$ implies that p divides ℓ . Furthermore both α and β are non-negative integers and β is maximal with this property. In particular we can write $N + \alpha \cdot p$ as a non-negative linear combination with elements from Z' . This gives $N + \alpha \cdot p = \sum_{p_j \in Z'} \kappa_j \cdot p_j$ for $\kappa_j \in \mathbb{N}$. Putting this together we have

$$\begin{aligned} f_v(n) &= (A^n)_{vv} \\ &= (A^{N+\alpha \cdot p + \beta \cdot \ell})_{vv} \\ &= (A^{\sum_{p_j \in Z'} \kappa_j \cdot p_j + \beta \cdot \ell})_{vv} \\ &\geq (A^{\beta \cdot \ell})_{vv} \cdot \prod_{p_j \in Z'} (A^{\kappa_j \cdot p_j})_{vv} \\ &\geq (A^\ell)_{vv}^\beta \cdot \prod_{p_j \in Z'} (A^{p_j})_{vv}^{\kappa_j} \\ &= f_v(\ell)^\beta \cdot \prod_{p_j \in Z'} f_v(p_j)^{\kappa_j} \\ &\geq f_v(\ell)^\beta \\ &\geq 2^\beta \\ &\geq 2^{\frac{n-N}{\ell} - 1} \in 2^{\Omega(n)} \end{aligned}$$

Note that no entry in A^n can grow faster than linearly exponential, so this lower bound directly induces a function $g_v \in 2^{\Theta(n)}$ rather than a function $g_v \in 2^{\Omega(n)}$.

In the graph setting this corresponds to repeating different walks from v to v of length ℓ for most of the walk, thus accumulating a linearly exponential amount of different walks and then finishing with some bounded amount of walks from v to v with lengths in Z' . $\blacktriangleleft$

► **Lemma 3.21.** *If $A \in \mathbb{N}^{k \times k}$, then each entry of A^n is an ultimately almost PORC function.*

Proof. We use the description of the diagonal entries of the matrices A_S^n given by Lemma 3.20 for all sets $S \subseteq [k]$ where A_S is the matrix A obtained by removing the rows and columns given by S . These correspond to the number of walks of length n from a vertex v to itself in the multigraph associated with the adjacency matrix A , restricted to the induced subgraph spanned by all the vertices not in S . For any $S \subseteq [k]$ we will index A_S the same way as we index A , i.e. $(A_S)_{ij} = A_{ij}$ whenever $i, j \in [k] \setminus S$. Lemma 3.20 in particular already proves this lemma for the diagonal entries.

Let $G = (V, E)$ with $V = [k]$ be the multigraph defined by the adjacency matrix A . Let W be any walk on G . Then we can obtain a unique simple path P_W from W by repeatedly

contracting cycles in W . We call two walks W_1 and W_2 equivalent if $P_{W_1} = P_{W_2}$. Note that this forms an equivalence relation on the walks on G and there is a bijection between paths on G and equivalence classes of walks on G and we can construct every walk in the equivalence class $\mathcal{W}_P$ corresponding to a path¹⁰ $P = v_1 \rightarrow \dots \rightarrow v_r$ as follows: At every vertex v_i we can add any walk from v_i back to v_i of any length ℓ_i , given that it doesn't pass through any of the vertices $v_1, \dots, v_{i-1}$. Note that this restriction is not necessary in general, but it ensures that every walk is uniquely constructed, a necessity for our proof. This construction gives a walk of length $r - 1 + \sum_{i=1}^r \ell_i$.

Fix some $u \neq v \in V$ and let $P = v_1 \rightarrow \dots \rightarrow v_r$ be any path from u to v , i.e. $v_1 = u$ and $v_r = v$. Then the class $\mathcal{W}_P$ contains exactly

$$f_P(n) := \sum_{(\ell_1, \dots, \ell_r)} \prod_{i=1}^r (A_{S_i}^{\ell_i})_{v_i v_i}$$

walks of length $n \in \mathbb{N}$ where the sum is over all $(\ell_1, \dots, \ell_r) \in \mathbb{N}^r$ with $n = r - 1 + \sum_{i=1}^r \ell_i$ and we define $S_i = \{v_1, \dots, v_{i-1}\}$. By Lemma 3.20 we know the structure of each of the $(A_{S_i}^{\ell_i})_{v_i v_i}$ as a function in ℓ_i . Call the v_i either 0-vertices, 1-vertices or exp-vertices respectively for each of the three cases of Lemma 3.20. Additionally denote by p_i the corresponding period for the 1-vertices and exp-vertices and in case of exp-vertices denote by N_i the smallest multiple of p_i , s.t. $(A_{S_i}^{N_i + j \cdot p_i})_{v_i v_i} > 0$ for all $j \in \mathbb{N}$. Define $I_0 := \{i \in [r] \mid v_i \text{ is a 0-vertex}\}$ and let I_1 and I_{exp} be defined analogously.

We first look at the case $I_{\text{exp}} = \emptyset$. Then $\prod_{i=1}^r (A_{S_i}^{\ell_i})_{v_i v_i} = 1$ iff $\ell_i = 0$ for all $i \in I_0$ and ℓ_i is a multiple of p_i for all $i \in I_1$, otherwise $\prod_{i=1}^r (A_{S_i}^{\ell_i})_{v_i v_i} = 0$. Thus $f_P(n)$ is given as the number of integer points in the polytope defined by $\sum_{i \in I_1} \ell_i \cdot p_i = n - r + 1$ and $\ell_i \geq 0$ for all $i \in I_1$. This is a scaled polytope with rational vertices and thus $f_P(n)$ is the Ehrhart quasi-polynomial (see [3, Chapter 3] for an introduction to Ehrhart Theory).

We now look at the case $I_{\text{exp}} \neq \emptyset$. Let v_e be one vertex with $e \in I_{\text{exp}}$ and let p be the gcd of all p_i with $i \in I_1 \cup I_{\text{exp}}$. This implies that $\prod_{i=1}^r (A_{S_i}^{\ell_i})_{v_i v_i} = 0$ whenever any of the ℓ_i with $i \in [r]$ are not a multiple of p and thus $f_P(n) = 0$ if $n \not\equiv_p r - 1$. Additionally by Bézout's lemma there is some $N \in \mathbb{N}$ which is a multiple of p s.t. that every number of the form $N + j \cdot p$ with $j \in \mathbb{N}$ can be written as a non-negative linear combination of the p_i with $i \in I_1 \cup I_{\text{exp}}$. Define $N' := N + r - 1 + \sum_{i \in I_{\text{exp}}} N_i$. Let $n = N' + j \cdot p$ for any $j \in \mathbb{N}$. We can write

$$n = N' + \underbrace{\frac{n - N' - \lfloor \frac{n - N'}{p_e} \rfloor \cdot p_e}{p}}_{\alpha :=} \cdot p + \underbrace{\left\lfloor \frac{n - N'}{p_e} \right\rfloor}_{\beta :=} \cdot p_e. \quad (2)$$

since $e \in I_{\text{exp}}$ implies that p divides p_e . Furthermore both α and β are non-negative integers and β is maximal with this property. In particular we can write $N + \alpha \cdot p$ as a non-negative linear combination with elements p_i with $i \in I_1 \cup I_{\text{exp}}$. This gives $N' + \alpha \cdot p = r - 1 + \sum_{i \in I_{\text{exp}}} N_i + \sum_{i \in I_1 \cup I_{\text{exp}}} \kappa_i \cdot p_i$ for $\kappa_i \in \mathbb{N}$. We choose $\ell_i = 0$ for $i \in I_0$ and $\ell_i = \kappa_i \cdot p_i$ for $i \in I_1$ and $\ell_i = N_i + \kappa_i \cdot p_i$ for $i \in I_{\text{exp}} \setminus \{e\}$ and $\ell_e = N_e + \kappa_e \cdot p_e + \beta \cdot p_e$. This describes a subset of the walks of length

$$r - 1 + \sum_{i \in [r]} \ell_i = r - 1 + \sum_{i \in I_{\text{exp}}} N_i + \sum_{i \in I_1 \cup I_{\text{exp}}} \kappa_i \cdot p_i + \beta \cdot p_e = n_1 + \alpha \cdot p + \beta \cdot p_e = n.$$

¹⁰ A path here is a sequence of edges, so in particular two paths with the same sequence of vertices do not have to be the same since G is a multigraph

In particular we have $(A_{S_i}^{\ell_i})_{v_i v_i} \geq 1$ for all $i \in [r]$ and $(A_{S_i}^{\ell_i})_{v_i v_i} \in 2^{\Theta(\ell_i)} = 2^{\Theta(n)}$ and thus $f_P(n) \geq 2^{\Omega(n)}$ for any $n \geq n_1$ with $n \equiv_p r - 1$. Note again that $f_P(n)$ can never grow faster than linearly exponential as it is bounded by the total number of walks through G , so $f_P(n)$ is of the form required by the statement of this lemma.

To finish the proof note that the class of ultimately almost PORC functions is closed under addition and multiplication. If we multiply a polynomial constituent with an exponential constituent, then either the polynomial constituent is identical to 0 or there is some $N \in \mathbb{N}$, s.t. the polynomial is strictly positive for every $n \geq N$. We have

$$(A^n)_{uv} = \sum_P f_P(n)$$

where the sum is over all paths P starting at u and ending at v . For any fixed $u, v \in V$ this is a sum of constantly many elements, bounded by $(k \cdot \max(A))^{k-1}$, where $\max(A)$ denotes the maximum entry of A . We conclude that $(A^n)_{uv}$ is an ultimately almost PORC function. $\blacktriangleleft$

► **Lemma 3.24.** *Let $\varphi(x_1, \dots, x_m) = \sum_{i=1}^r a_i \cdot \prod_{j=1}^m \binom{x_j}{d_{i,j}}$ with $a_i \in \mathbb{Z}$ and the coefficients of the dominating terms being positive. Then there are $a'_1, \dots, a'_{r'}$ in $\mathbb{N}$ and $c_1, \dots, c_{r'}$ in $\mathbb{N}$ with $\varphi(x_1, \dots, x_m) = \sum_{i=1}^{r'} a'_i \cdot \prod_{j=1}^m \binom{x_j - c_i}{d_{i,j}}$.*

Proof. Let D_φ be the set of dominating terms of φ . We consider a set of dominating terms D to be smaller than another set of dominating terms D' , whenever $D \neq D'$ and all terms of in D are dominated by terms in D' . We prove the statement via induction over D . In case $D_\varphi = \emptyset$ we have $\varphi = 0$ which is trivially a functional closure property. So let $D_\varphi \neq \emptyset$ and let $a \cdot \binom{x_1}{d_1} \cdots \binom{x_m}{d_m}$ be any term from D_ψ . We use a similar idea to Lemma 3.16 and replace the binomial basis by a shifted binomial basis. Consider the terms $a_i \cdot \binom{x_1}{d_1} \cdots \binom{x_{i-1}}{d_{i-1}} \binom{x_i}{d_i-1} \binom{x_{i+1}}{d_{i+1}} \cdots \binom{x_m}{d_m}$ in ψ for all $i \in [m]$ with $d_i \geq 1$. These are precisely the (potentially new) dominating terms of $\varphi - a \cdot \binom{x_1}{d_1} \cdots \binom{x_m}{d_m}$. In order to ensure the positivity of the a_i we instead consider $\psi := \varphi - a \cdot \binom{x_1-c}{d_1} \cdots \binom{x_m-c}{d_m}$ for some $c \in \mathbb{N}$ with $c > -\frac{a_i}{a}$ for all $i \in [m]$ with $d_i \geq 1$. We recall that the Chu-Vandermonde identity gives us $\binom{x-c}{r} = \sum_{i=0}^r (-1)^{r-i} \binom{r-i+c-1}{r-i} \binom{x}{i}$. Thus the new coefficient a'_i of $\binom{x_1}{d_1} \cdots \binom{x_{i-1}}{d_{i-1}} \binom{x_i}{d_i-1} \binom{x_{i+1}}{d_{i+1}} \cdots \binom{x_m}{d_m}$ in ψ' is $a_i + a \cdot (-c) > 0$. The new coefficient of $\binom{x_1}{d_1} \cdots \binom{x_m}{d_m}$ in ψ is zero and thus D_ψ is smaller than D_φ . By induction we know that ψ can be written as a positive linear combination of products of shifted binomials and thus $\varphi = \psi + a \cdot \binom{x_1-c}{d_1} \cdots \binom{x_m-c}{d_m}$ has the same property. $\blacktriangleleft$

► **Lemma 3.25.** *Let $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ be a multivariate polynomial with rational coefficients, such that whenever ψ is formed from φ by replacing any set of variables – including the empty subset – by constants from $\mathbb{N}$, then all dominating terms of ψ have positive coefficients. Then φ is a functional closure property of #FA.*

Proof. We prove the statement via induction over the number of variables m . If $m \leq 1$, then φ is a constant polynomial and thus a functional closure property of #FA by Lemma 3.17. If $m \geq 2$, and let $f_1, \dots, f_m \in \text{\#FA}$. Write φ as a positive integer linear combination of products of shifted binomials using Lemma 3.24 and let $c \in \mathbb{N}$ be the maximum occurring shift. We now distinguish between the f_i assuming values from $\{0, \dots, c-1\}$ and them assuming values of at least c and all combinations thereof for the different f_i . Using the fact

that all subtractions subtract at most a value of c we can replace the subtractions $x_i - c'$ by $\max(x_i - c', 0)$ without changing the value for $x_i \geq c$ and get the following:

$$\begin{aligned} \psi(f_1(w), \dots, f_m(w)) &= \left(\prod_{i \in [m]} \mathbb{1}_{f_i(w) \geq c} \right) \cdot \varphi_{\text{clamped}}(f_1(w), \dots, f_m(w)) \\ &+ \sum_{\emptyset \subsetneq I \subseteq [m]} \sum_{\rho} \left(\prod_{i \in I} \mathbb{1}_{f_i(w) = \rho(i)} \right) \cdot \left(\prod_{i \in [m] \setminus I} \mathbb{1}_{f_i(w) \geq c} \right) \cdot \psi_{I, \rho}(f_1(w), \dots, f_m(w)) \end{aligned}$$

where ρ sums over all functions $I \rightarrow \{0, \dots, c-1\}$, φ_{clamped} is φ , but with all subtractions $x_i - c'$ replaced by $\max(x_i - c', 0)$ and $\varphi_{I, \rho}$ is φ , but all variables x_i with $i \in I$ are replaced by the constants $\rho(i)$. The $\varphi_{I, \rho}$ are non-negative integer-valued multivariate polynomials in fewer variables and when replacing any variables by further constants we only obtain polynomials that can also be obtained by replacing variables directly in φ . Thus by induction the $\varphi_{I, \rho}$ are a functional closure property of $\#FA$ and $\varphi(f_1, \dots, f_m) \in \#FA$ by Lemmas 3.1, 3.2, 3.9, 3.11, and 3.15. $\blacktriangleleft$

► **Lemma 3.26.** *A multivariate polynomial $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ with rational coefficients is a functional closure property of $\#FA$ iff for every ψ that can be formed from φ by replacing any subset of variables – including the empty set – by constants from $\mathbb{N}$, then all dominating terms of ψ have positive coefficients.*

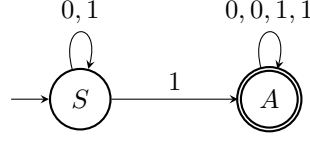
Proof. Lemma 3.25 already proves that all multivariate polynomials of this form are functional closure properties of $\#FA$.

Now let $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ be a multivariate polynomial with rational coefficient and a functional closure property of $\#FA$. By Theorem 3.23 φ can be written as a finite sum of finite products of ultimately PORC functions. W.l.o.g. let all quasiperiods be the same $p \in \mathbb{N}$ and all offsets be the same $N \in \mathbb{N}$ and a multiple of p , for example by choosing the lowest common multiple of the quasiperiods and choosing the lowest multiple of p that is bigger than all offsets. Then

$$\varphi(x_1, \dots, x_m) = \sum_{i=1}^r \prod_{j=1}^m \varphi_0^{(i,j)}(x_j) \quad (3)$$

where $\varphi_0^{(i,j)}$ is an ultimately PORC function with constituents $\varphi_0^{(i,j)}, \dots, \varphi_{p-1}^{(i,j)}$. Consider inputs on a grid of the form $(N + i_1 \cdot p, N + i_2 \cdot p, \dots, N + i_m \cdot p)$ for $i_1, \dots, i_m \in \mathbb{N}$. For all these inputs each of the ultimately PORC functions always use the same constituents, namely the $\varphi_0^{(i,j)}$. Thus φ agrees with $\varphi'(x_1, \dots, x_m) := \sum_{i=1}^r \prod_{j=1}^m \varphi_0^{(i,j)}(x_j)$ on all inputs from this grid. Furthermore the grid is infinite in all dimensions and thus $\varphi = \varphi'$ by multivariate polynomial interpolation (see [11] for an exposition of multivariate polynomial interpolation). W.l.o.g. assume that $\varphi_0^{(i,j)}$ is not the constant zero function. Then it has a positive leading coefficient, otherwise $\varphi_0^{(i,j)}$ would not be a PORC function. Combined with the fact that each of the $\varphi_0^{(i,j)}$ is univariate the product $\prod_{j=1}^m \varphi_0^{(i,j)}(x_j)$ for a fixed $i \in [r]$ has exactly one dominating term when written in the binomial basis. Its coefficient is precisely the product of the leading coefficients of the $\varphi_0^{(i,j)}$ and thus positive. As a result all dominating terms in the binomial basis of φ' (and thus φ) have positive coefficients, since no cancellations can occur unless the dominating term of a summand dominates the dominating term of another summand, in which case the smaller term would no longer be a dominating term.

Note that whenever any variable, for example x_j , in φ is replaced by a constant c , then this only changes each function $\varphi_0^{(i,j)}(x_j)$ and thus $\varphi_0^{(i,j)}(x_j)$ to be the constant $\varphi_0^{(i,j)}(c)$ which is non-negative. $\blacktriangleleft$



■ **Figure 2** NFA computing the binary the value of its input interpreted as a binary number. Edges with a multiplicity of 2 are denoted by listing the edge label twice.

A.3 Missing proofs from section 4

► **Lemma 4.3** (Folklore). *The function $f : \{0, 1\}^* \rightarrow \mathbb{N}$ defined by being the value of $w \in \{0, 1\}^*$ interpreted as a binary number is in #FA. Additionally it is possible to extend the domain of f to any alphabet $\Sigma \supseteq \{0, 1\}$ where the value of f is determined while ignoring any symbols not in $\{0, 1\}$.*

Proof. We define

$$a = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad A_\sigma = \begin{pmatrix} 1 & \sigma \\ 0 & 2 \end{pmatrix}, \quad \text{and} \quad b = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

A visualization of the corresponding NFA M via Lemma 3.19 can be found in Figure 2. We prove that M computes f by showing via induction that for every $w \in \{0, 1\}^*$ with $|w| = n$ we have

$$\prod_{i=1}^n A_{w_i} = \begin{pmatrix} 1 & f(w) \\ 0 & 2^n \end{pmatrix}.$$

For $n = 0$ this is obviously correct as $f(\varepsilon) = 0$. For the induction step consider the word $w\sigma$ of length $n + 1$ for $\sigma \in \{0, 1\}$. Then

$$\left(\prod_{i=1}^n A_{w_i} \right) \cdot A_\sigma = \begin{pmatrix} 1 & f(w) \\ 0 & 2^n \end{pmatrix} = \begin{pmatrix} 1 & \sigma + 2f(w) \\ 0 & 2^{n+1} \end{pmatrix} = \begin{pmatrix} 1 & f(w\sigma) \\ 0 & 2^{n+1} \end{pmatrix}$$

and we are done. Extending the alphabet and ignoring those new symbols can be done by adding a self-loop to each state of M . This corresponds to an identity matrix and thus doesn't influence the result. ◀

► **Lemma 4.2.** *Let $c_0, \dots, c_N \in \mathbb{N}$. Then there is an integer-valued polynomial $q : \mathbb{Q} \rightarrow \mathbb{Q}$ with $q(n) = c_n$ for all $n \in \{0, \dots, N\}$ and $q(n') \geq 0$ for all $n' \in \mathbb{N}$.*

Proof. We inductively construct integer-valued polynomials $q_0, \dots, q_N$, that each interpolate one value more. We start by setting $q_0(x) := c_0 \cdot \binom{x}{0}$. Clearly $q_0(0) = c_0$. We proceed inductively by setting $q_{i+1}(x) := q_i(x) + (c_{i+1} - q_i(i+1)) \cdot \binom{x}{i+1}$. Note that $\binom{n}{i+1} = 0$ for $n \in \{0, \dots, i\}$ and thus q_{i+1} and q_i agree on all such n . The final q is then constructed as $q(x) := q_N(x) + \alpha \cdot \binom{x}{N+1}$, for some big enough $\alpha \in \mathbb{N}$, s.t. $q(n') \geq 0$ for all $n' \in \mathbb{N}$. Such an α always exists since $\binom{x}{N+1}$ has higher degree than q_N . ◀

► **Theorem 4.6.** *Let $S \subseteq \mathbb{N}^m$ admit polynomial cluster sequences. Then any function $\varphi : \mathbb{N}^m \rightarrow \mathbb{N}$ is a functional promise closure property of #FA with regard to S iff there is a functional closure property $\psi : \mathbb{N}^m \rightarrow \mathbb{N}$ of #FA with $\varphi|_S = \psi|_S$.*

Proof. If such a φ' exists, we directly see that φ is a functional promise closure property of $\#FA$ with respect to S . Indeed for any $f_1, \dots, f_m \in \#FA$ we construct $g = \varphi' \circ (f_1, \dots, f_m) \in \#FA$ and see $g(w) = \varphi(f_1(w), \dots, f_m(w))$ for every $w \in \Sigma^*$ with $(f_1(w), \dots, f_m(w)) \in S$.

Now on the other hand let φ be any functional promise closure property of $\#FA$ with regard to S . Again define the alphabet $\Sigma = \{\sigma_1, \dots, \sigma_m\}$ and the functions $f_i : \Sigma^* \rightarrow \mathbb{N}$ where $f_i(w) := \#_i(w)$ is defined as the number of occurrences $\#_i(w)$ of the symbol σ_i in w . Applying the closure property to $f_1, \dots, f_m$ gives that there is some $g \in \#FA$ with $g(w) = \varphi(f_1(w), \dots, f_m(w))$ for all $w \in \Sigma^*$ with $(f_1(w), \dots, f_m(w)) \in S$. Let $M = (Q, \Sigma, \text{wt}, \text{in}, \text{out})$ be an NFA computing g . This induces transition matrices $A_\sigma \in \mathbb{N}^{|Q| \times |Q|}$ for each symbol $\sigma \in \Sigma$ and vectors $a, b \in \mathbb{N}^{|Q|}$, s.t. $a^T \prod_{j=1}^{|w|} A_{w_j} b = g(w)$ for all $w \in \Sigma^*$. Restricting to words of the form $w = \sigma_1^{n_1} \sigma_2^{n_2} \dots \sigma_m^{n_m}$ for $n_1, \dots, n_m \in S$ gives $a^T (\prod_{i=1}^m A_{\sigma_i}^{n_i}) b = \varphi(n_1, \dots, n_m)$. Using Lemma 3.21 on each of the $A_{\sigma_i}^{n_i}$ we see that every entry of $A_{\sigma_i}^{n_i}$ is an ultimately almost PORC function in n_i . Consequently every entry of $\prod_{i=1}^m A_{\sigma_i}^{n_i}$ is a finite sum of products of different ultimately almost PORC functions and the same holds for $\chi(n_1, \dots, n_m) := a^T (\prod_{i=1}^m A_{\sigma_i}^{n_i}) b$. Note that $\chi|_S = \varphi|_S$.

Remains to show that we can rewrite χ as some ψ with $\psi|_S = \chi|_S$, s.t. none of the constituents are exponential. Again we do this by looking at each summand individually and replacing one exponential constituent at a time. For this let $\chi^{(1)}(n_1) \dots \chi^{(m)}(n_m)$ be one of the summands of χ where $\chi^{(1)}, \dots, \chi^{(m)}$ are all ultimately almost PORC functions, with periods $p_1, \dots, p_m$, offsets $N_1, \dots, N_m$ and constituents $\chi_0^{(i)}, \dots, \chi_{p_i-1}^{(i)}$ for each $i \in [m]$. W.l.o.g. choose the offsets big enough, such that every constituent $\chi_j^{(i)}$ is either constant zero for $n_i \geq N_i$ or never zero for any $n_i \geq N_i$. Let $\chi_j^{(i)}$ be one of the exponential constituents and let $\gamma \in \mathbb{R}^+$ and $N \in \mathbb{N}$ be, s.t. $\chi_j^{(i)}(n_i) \geq 2^{\gamma n_i}$ for $n_i \geq N$. Let $B_{\max, i}$ be the maximum the i -th coordinate assumes in any preimage of a singleton set under τ where the i -th coordinate is bounded. We replace $\chi_j^{(i)}$ with a polynomial that interpolates $\chi^{(i)}$ for all values of at most $B_{\max, i}$. Lemma 4.2 ensures this is possible with an integer-valued polynomial that is non-negative for all of $\mathbb{N}$. Call the resulting functions after replacement $\psi^{(i)}$ and $\psi_j^{(i)}$. We claim that this does not affect the value of the product for any $(n_1, \dots, n_m) \in S$. For the sake of contradiction the value of the product did change for some $(c_1, \dots, c_m) \in S$. We thus have

$$\chi^{(1)}(c_1) \dots \chi^{(m)}(c_m) \neq \chi^{(1)}(c_1) \dots \chi^{(i-1)}(c_{i-1}) \cdot \psi^{(i)}(c_i) \cdot \chi^{(i+1)}(c_{i+1}) \dots \chi^{(m)}(c_m)$$

which implies $\chi^{(1)}(c_1) \dots \chi^{(i-1)}(c_{i-1}) \cdot \chi^{(i+1)}(c_{i+1}) \dots \chi^{(m)}(c_m) \neq 0$ and $c_i \geq N_i$ as no function except $\chi^{(i)}$ for inputs at least as big as N_i has been changed. By the same reason we know $\chi_j^{(i)}(c_i) \neq \psi_j^{(i)}(c_i)$ and $c_i \equiv_{p_i} j$.

Consider the preimage $T = \tau^{-1}(c_1 \text{ srem}_{N_1} p_1, \dots, c_m \text{ srem}_{N_m} p_m)$. Clearly $(c_1, \dots, c_m) \in T$. The i -th coordinate of T has to be unbounded, because $c_i \leq B_{\max, i}$ would imply $\psi_j^{(i)}(c_i) = \chi^{(i)}(c_i) = \chi_j^{(i)}(c_i)$ by the construction of $\psi_j^{(i)}$. Furthermore the behaviour of each of the $\chi^{(k)}$ is only dictated by at most one of the constituents each. By our choice of the offsets we have $\chi^{(k)}(n_k) \neq 0$ for all $k \neq i$ and all $(n_1, \dots, n_m) \in T$. Since S admits polynomial cluster sequences there is a polynomial cluster sequence $T' \subseteq T$ with regards to dimension i with polynomial $q : \mathbb{N} \rightarrow \mathbb{N}$.

We construct m functions $f'_k \in \#FA$ over the alphabet $\Sigma = \{\sigma_{1,0}, \sigma_{1,1}, \dots, \sigma_{m,0}, \sigma_{m,1}\}$ with $2m$ symbols, by using Lemma 4.3 on the symbols $\sigma_{k,0}$ and $\sigma_{k,1}$ each while ignoring any of the other symbols. This allows us to encode any $(n_1, \dots, n_m) \in T'$ as independent binary representations without leading zeros for each n_k into a string of total length $\sum_{k=1}^m \lceil \log_2(n_k) + 1 \rceil \leq m \cdot \lceil \log_2(q(n_i)) + 1 \rceil$ which is in $O(\log(n_i))$ for fixed m and q .

Now let $w \in \Sigma^*$ be any such string corresponding to a value $(n_1, \dots, n_m) \in T'$. Combining all of this we again reach a contradiction to the fact that NFAs can only compute at most linearly exponential functions via

$$\begin{aligned} \varphi(f'_1(w), \dots, f'_m(w)) &\geq \chi^{(1)}(c_1) \cdots \chi^{(i-1)}(c_{i-1}) \cdot \chi^{(i)}(f'_i(w)) \cdot \chi^{(i+1)}(c_{i+1}) \cdots \chi^{(m)}(c_m) \\ &\geq \chi^{(i)}(f'_i(w)) \\ &= \chi_j^{(i)}(f'_i(w)) \\ &= \chi_j^{(i)}(n_i) \\ &\geq 2^{n_i}. \end{aligned}$$

Note that $|w| \in O(\log(n_i))$ implies 2^{n_i} is doubly exponential in the input length. $\blacktriangleleft$

► **Lemma 4.8.** *Let $S \subseteq \mathbb{Q}^m = \mathbb{Q}^{j+k}$ be a monotone graph variety. Then the set $S \cap \mathbb{N}^m = \mathbb{N}^{j+k}$ admits polynomial cluster sequences.*

Proof. Let $N_1, \dots, N_m \in \mathbb{N}$ and $p_1, \dots, p_m \in \mathbb{N}$ be fixed and consider the preimage T of any singleton $(c_1, \dots, c_m) \in \{0, \dots, N_1 + p_1 - 1\} \times \dots \times \{0, \dots, N_m + p_m - 1\}$ under the corresponding shifted grid projection τ of S . Let $(n_1, \dots, n_m) \in T$ and let $i \in [m]$ be arbitrary. If the i -th coordinate in T is bounded then we are done. If the i -th coordinate in T is unbounded we distinguish two cases.

In case $1 \leq i \leq j$, we choose the T' to be the part of S spanned by choosing the values of the free variables as $\{(n_1, \dots, n_{i-1}, n_i + \kappa \cdot pd, n_{i+1}, \dots, n_j) \mid \kappa \in \mathbb{N}\}$ where p is the lcm of $p_1, \dots, p_m$ and d is the common denominator of all of the $\mu_1, \dots, \mu_k$. Clearly the i -th coordinate of T' is unbounded and T' is infinite. Further since all other free variables are constant, each dependent variable is a monotone univariate polynomial in the i -th coordinate and thus the sum of all coordinates is bounded by some polynomial in the i -th coordinate. Remains to show $T' \subseteq T$. For this let $(n'_1, \dots, n'_m) \in T'$. Clearly $n'_\ell \text{ srem}_{N_\ell} p_\ell = n_\ell \text{ srem}_{N_\ell} p_\ell$ for $\ell \in [j] \setminus \{i\}$, since $n'_\ell = n_\ell$. Additionally $n'_i \text{ srem}_{N_i} p_i = n_i \text{ srem}_{N_i} p_i$, since $n'_i = n_i + \kappa \cdot pd$ and $n_i \geq N_i$. Furthermore $n'_\ell \text{ srem}_{N_\ell} p_\ell = n_\ell \text{ srem}_{N_\ell} p_\ell$ for $\ell \in [m] \setminus [j]$, because either $n'_\ell = n_\ell$ or the ℓ -th coordinate is unbounded and thus $n'_\ell \geq n_\ell \geq N_\ell$ and $n'_\ell \equiv_{p_\ell} n_\ell$ since n'_ℓ is dependent on n'_i as a univariate polynomials and for any univariate polynomial $q(x)$ we have $q(x) \equiv_{p_\ell} q(x + \delta)$ if δ is a multiple of $p_\ell \cdot h$. Thus $(n'_1, \dots, n'_m) \in T$.

Remains to consider the case where $i > j$. Since the i -th coordinate in T is unbounded, n_i is a function depending on at least one of the free variables $n_{i'}$. We repeat the previous argument by continuing with i' instead of i which finishes the proof, since $n_i \geq \alpha \cdot n_{i'}$ for some $\alpha \in \mathbb{R}^+$. $\blacktriangleleft$

► **Theorem 4.9.** *Let $S \subseteq \mathbb{Q}^m = \mathbb{Q}^{j+k}$ be a monotone graph variety and let $I = I(S)$ be its vanishing ideal. A multivariate polynomial $\varphi : S \rightarrow \mathbb{N}$ is a functional promise closure property of $\#FA$ with regard to S if and only if there exists a $\psi \in I$ such that $\varphi + \psi$ is a multivariate functional closure property of $\#FA$.*

Proof. If there exists a $\psi \in I$ such that $\varphi + \psi$ is a multivariate functional closure property of $\#FA$, then $(\varphi + \psi)|_S = \varphi|_S$ and thus φ is a functional promise closure property of $\#FA$ with regard to S by definition. On the other hand if φ is a functional promise closure property of $\#FA$ with regard to S , then there is a functional closure property φ' of $\#FA$ with $\varphi|_S = \varphi'|_S$. Note that a priori φ' might not be a polynomial, but just a finite sum of finite products of univariate ultimately PORC functions.

Wlog all thresholds N and quasiperiods p of φ' coincide. We write $p^{\times m} = (p, p, \dots, p)$ and $N^{\times m} = (N, N, \dots, N)$. For each $\vec{i} \in \{0, \dots, p + N - 1\}^m$ we have a polynomial $\varphi'_{\vec{i}}$, and $\varphi'(\vec{n}) = \varphi'_{\vec{n} \bmod N^{\times m} p}(\vec{n})$ for all $\vec{n} \in \mathbb{N}^m$. Define the set $S' = \vec{\mu}^{-1}(\mathbb{N}^m) \subseteq \mathbb{N}^j$ as all $\vec{s} \in \mathbb{N}^j$, s.t. $\mu_a(\vec{s}) \in \mathbb{N}$ for all $a \in [k]$. Let $\tilde{\varphi}'_{\vec{i}} : \mathbb{Q}^j \rightarrow \mathbb{Q}$ denote the pullback via μ , i.e., $\tilde{\varphi}'_{\vec{i}}(s_1, \dots, s_j) = \varphi'_{\vec{i}}(s_1, \dots, s_j, \mu_1(\vec{s}), \dots, \mu_k(\vec{s}))$. Analogously define the polynomial $\tilde{\varphi} : S' \rightarrow \mathbb{Q}$. Since φ' is only defined on $\mathbb{N}^m$, the definition of its pullback requires a bit more care, but since the pullback of φ' is defined for all of S' , we can then analogously define $\tilde{\varphi}' : S' \rightarrow \mathbb{Q}$. Observe that $\tilde{\varphi}$ and $\tilde{\varphi}'$ coincide on S' . Let d be the common denominator of $\mu_1, \dots, \mu_k$. Note that for any fixed $a \in [k]$ and $\vec{\sigma} \in S'$ we have that $\mu_a(\vec{\sigma} + dp\vec{b}) \bmod p$ is constant for all $\vec{b} \in \mathbb{N}^j$. In particular we also have $\vec{\sigma} + dp\vec{b} \in S'$. Due to the monotonicity of each of the μ_a , for big enough $\vec{b}$, $\mu_a(\vec{\sigma} + dp\vec{b})$ will either always be at least N or μ_a is a constant function. Thus there is now some $\vec{\sigma} \in S'$ with $\vec{\sigma} \geq N^{\times j}$, such that even $\mu_a(\vec{\sigma} + dp\vec{b}) \bmod p$ is constant for all $\vec{b} \in \mathbb{N}^j$. Let $\vec{\theta} = (\vec{\sigma}, \mu_1(\vec{\sigma}), \dots, \mu_k(\vec{\sigma})) \bmod N^{\times m}$. Hence, for all $\vec{b} \in \mathbb{N}^j$ we have $\tilde{\varphi}'(\vec{\sigma} + dp\vec{b}) = \tilde{\varphi}'_{\vec{\theta}}(\vec{\sigma} + dp\vec{b})$. Since $\tilde{\varphi}$ and $\tilde{\varphi}'$ coincide on S' , it follows that $\tilde{\varphi}$ and $\tilde{\varphi}'_{\vec{\theta}}$ coincide on an infinitely large full-dimensional subgrid (namely on the grid formed by $\vec{\sigma} + dp\vec{b}$ for $b \in \mathbb{N}^j$). Hence by multivariate polynomial interpolation, $\tilde{\varphi} = \tilde{\varphi}'_{\vec{\theta}}$. Hence, $\varphi|_S = \varphi'_{\vec{\theta}}|_S$. Their difference $\varphi'_{\vec{\theta}} - \varphi$ is the desired polynomial $\psi \in I$. It remains to show that $\varphi'_{\vec{\theta}}$ is a multivariate closure property of #FA. Repeating the same argument as in Lemma 3.26 to prove that the dominating terms of $\varphi'_{\vec{\theta}}$ have positive coefficients, even after replacing some variables by constants. Using Lemma 3.26 again, shows that $\varphi'_{\vec{\theta}}$ is a multivariate closure property of #FA. $\blacktriangleleft$