Monotonicity Characterizations of Regular Languages

Yoav Feinstein
School of Engineering and Computer Science, Hebrew University, Jerusalem, Israel

Orna Kupferman
School of Engineering and Computer Science, Hebrew University, Jerusalem, Israel

Abstract

Each language $L \subseteq \Sigma^*$ induces an infinite sequence $\{Pr(L,n)\}_{n=1}^{\infty}$, where for all $n \geq 1$, the value $Pr(L,n) \in [0,1]$ is the probability of a word of length n to be in L, assuming a uniform distribution on the letters in Σ. Previous studies of $\{Pr(L,n)\}_{n=1}^{\infty}$ for a regular language L, concerned zero-one laws, density, and accumulation points. We study monotonicity of $\{Pr(L,n)\}_{n=1}^{\infty}$, possibly in the limit. We show that monotonicity may depend on the distribution of letters, study how operations on languages affect monotonicity, and characterize classes of languages for which the sequence is monotonic. We extend the study to languages L of infinite words, where we study the probability of lasso-shaped words to be in L and consider two definitions for $Pr(L,n)$. The first refers to the probability of prefixes of length n to be extended to words in L, and the second to the probability of word w of length n to be such that w^ω is in L. Thus, in the second definition, monotonicity depends not only on the length of w, but also on the words being periodic.

2012 ACM Subject Classification Theory of computation $\rightarrow$ Formal languages and automata theory

Keywords and phrases Regular Languages, Probability, Monotonicity, Automata

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2023.26

Funding Supported by the Israel Science Foundation, Grant 2357/19, and by the European Research Council, Advanced Grant ADVANSYNT.

1 Introduction

Consider an alphabet Σ, and assume that letters in Σ are drawn uniformly at random. The probability of a random word of length n to be in a given language $L \subseteq \Sigma^*$ is then $Pr(L,n) = \frac{|\{w \in \Sigma^n | w \in L\}|}{|\Sigma^n|}$. Thus, each language L induces an infinite sequence $\{Pr(L,n)\}_{n=1}^{\infty}$ of values in $[0,1]$. The sequence describes how the length of words influences the probability of their membership in the language.

Several studies in *finite-model theory* refer to the asymptotic behavior of models satisfying a given property. The most known studies in this direction concern *zero-one laws* for different specification formalisms. For example, a zero-one law for first-order sentences states that for every property ψ expressible in first-order logic, the probability of finite structures that are drawn uniformly at random to satisfy ψ tends to 0 or 1 when the size of the structure tends to ∞ [10, 8]. For regular languages, an analogue zero-one law would state that the sequence $Pr(L,n)$ tends to 0 or 1. It is easy to see that regular languages, even unary ones, do not respect a zero-one law. For example, the language $L = (aa)^*$ over $\Sigma = \{a\}$ contains exactly all words of even length, and so $\{Pr(L,n)\}_{n=1}^{\infty}$ alternates between 0 and 1. In [19], Sin'ya characterized regular languages whose asymptotic probability converges to 0 or 1, and described a linear-time algorithm for deciding whether the language of a given deterministic automaton has a zero-one behavior.

Another study of the asymptotic behavior of $\{Pr(L,n)\}_{n=1}^{\infty}$ concerns *accumulation points* of the sequence, namely points to which a subsequence converges to. As shown in [2, 18], when L is regular, there are only finitely many such points, and they are all rational. The

43rd IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2023).
Editors: Patricia Bouyer and Srikanth Srinivasan; Article No. 26; pp. 26:1–26:19
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

above works also study the *density* of L, which is the limit of $\{Pr(L,n)\}_{n=1}^{\infty}$. A Markov-chain based approach to reasoning about the density of a regular language is presented in [4], which describes a cubic algorithm for calculating the density (or determine that one does not exist) of a language given by a deterministic automaton. Finally, the limit of the sequence $\frac{1}{n}\sum_{i\in\{1,\ldots,n\}} Pr(L,i)$, is studied in [3], and its convergence is related to that of $\{Pr(L,n)\}_{n=1}^{\infty}$.

In this paper we study the *monotonicity* of $\{Pr(L,n)\}_{n=1}^{\infty}$. We say that L is *eventually monotonic* if there is $m \geq 1$ such that for all $n \geq m$, we have that $Pr(L,n+1) \geq Pr(L,n)$ (monotonically non-decreasing) or for all $n \geq m$, we have that $Pr(L,n+1) \leq Pr(L,n)$ (monotonically non-increasing). When $m = 1$, the sequence is *monotonic*, and when the probability is strictly increased or decreased, we say that the sequence is monotonically increasing or monotonically decreasing. Let us consider again the language $L = (aa)^*$, yet assume it is defined over the alphabet $\Sigma = \{a,b\}$. Now, $\{Pr(L,n)\}_{n=1}^{\infty}$ does tend to 0. Indeed, $Pr(L,n) = \frac{1}{2^n}$ for even n's, and is 0 for odd n's. On the other hand, $\{Pr(L,n)\}_{n=1}^{\infty}$ is *never monotonic*, as for every $n \geq 1$, we have that $Pr(L,2n) > Pr(L,2n+1)$ yet $Pr(L,2n+1) < Pr(L,2n+2)$. Thus, a language L may have a zero-one behavior and not be eventually monotonic. Implication in the other direction does not hold either. For example, the language $L = b\cdot(a+b)^* + a^*$ has $Pr(L,n) = \frac{1}{2} + \frac{1}{2^n}$. Thus, L is monotonically decreasing, yet it tends to $\frac{1}{2}$, and does not have a zero-one behavior.

We start with some theoretical properties of monotonicity of regular languages. Recall that we define $Pr(L,n)$ with respect to a uniform distribution on the alphabet. We show that the probability according to which letters are drawn may actually change the monotonicity characteristic of languages, even in the limit. This is in contrast with zero-one laws for regular languages, which are independent of the distribution (as long as all letters have a positive probability). We study the sensitivity of monotonicity to Boolean operations. It is easy to come up with languages with dual monotonicity whose union and intersections are not monotonic. We show that the union and intersection of languages that are both increasing or both decreasing need not be monotonic, even eventually. We then consider the case of unary languages, thus when $\Sigma = \{a\}$. There, $Pr(L,n) \in \{0,1\}$, and it is easy to characterize monotonic languages. We continue and point to positive cases, namely classes of monotonic languages. For example, we show that $\{Pr(L,n)\}_{n=1}^{\infty}$ is eventually monotonic when L is recognizable by a *deterministic weak automaton* of *depth* or *width* 1. The characterization is tight, in the sense that removing one of these limitations, one may end up in a language that is never monotonic. A different tight characterization we give is of 2-state *counter-free* deterministic automata. Our analysis is based on results from linear algebra regarding the *stochastic matrix* induced by the automata, and it is valid also with respect to non-uniform distributions of the alphabet.

Moving to languages of infinite words, we consider lasso-shaped words, and study three sequences. The first two, denoted $\{Pr^{\exists}(L,n)\}_{n=1}^{\infty}$ and $\{Pr^{\forall}(L,n)\}_{n=1}^{\infty}$, refer to the probability of prefixes of length n to be extendable, by some or all suffixes, respectively, to words in L. We show that $Pr^{\exists}(L,n)$ and $Pr^{\forall}(L,n)$ are related to the probability of words of length n to be good prefixes for L and bad prefixes for the complement of L, respectively, implying the monotonicity of the sequences. The third sequence is $\{Pr^{\omega}(L,n)\}_{n=1}^{\infty}$, where $Pr^{\omega}(L,n)$ is the probability of a word w of length n to be such that w^{ω} is in L. Thus, here, monotonicity depends also on the words being periodic. For example, while it is easy to see that the language of finite prefixes of $(aab)^*$ is monotonically decreasing, with $Pr(L,n) = \frac{1}{2^n}$, we have that $Pr^{\omega}((aab)^{\omega},n)$ is never monotonic, as it is 0 for n's that are not multiplications of 3.

We describe a construction that transforms a deterministic parity automaton $\mathcal{A}$ to a deterministic automaton for all finite words u such that $u^\omega \in L(\mathcal{A})$. The construction is exponential, and we prove a matching lower bound. Using the construction, we are able to lift some of the positive results about languages of finite words to the setting of infinite words. We also discuss the characterizations when the languages are given by formulas in LTL (or LTL$_f$, for the case of finite words) [15, 9].

Beyond the theoretical interest, properties of $\{Pr(L,n)\}_{n=1}^{\infty}$ are useful when reasoning about L. We give here some examples. In the context of decision problems about regular languages, researchers suggested approximated algorithms that refer to asymptotic behavior. For example, [14] studies *almost equivalence* of regular languages, where two languages $L_1, L_2 \subseteq \Sigma^*$ are almost equivalent if $Pr(L_1 \triangle L_2, n)$ tends to 0, where $L_1 \triangle L_2$ denotes the symmetric difference between L_1 and L_2. General monotonicity properties of $L_1 \triangle L_2$ indicate how L_1 and L_2 differ from each other in the limit. Similar reasoning can be made about intersection of languages, their union, and more.

Our study of monotonicity for languages of infinite words is motivated by the need to sort results of vacuity checks in model checking. *Vacuity* detection is a method for finding errors in the model-checking process when the specification is found to hold in the model. Most vacuity algorithms are based on checking the effect of applying mutations on the specification [1]. It has been recognized that while in many cases vacuity results are valued as highly informative, there are also cases in which the results are viewed as meaningless by users [17, 5]. In [7], the authors suggested to rank vacuity results based on the probability of the mutated specification to hold in a random computation. For example, two natural mutations of the specification $G(\mathit{req} \rightarrow F\mathit{ready})$ are $G(\neg \mathit{req})$ and $GF\mathit{ready}$. It is agreed that satisfaction of the first mutation is more alarming than satisfaction of the second. The methodology explains this by the probability of $G(\neg \mathit{req})$ to hold in a random computation being 0, whereas the probability of $GF\mathit{ready}$ being 1. The above definition assumes an infinite computation. As discussed in [7], in the context of model checking, it is more relevant to refer to the probability of the mutation to hold in computations in finite-state systems. Specifically, it is suggested in [7] to study $Pr(L,k,l)$, which is the probability that a lasso-shaped word with a prefix of length k and a loop of length l, belongs to L. Our study of lasso-shaped words does this for the special cases $l = \infty$ or $k = 0$. In particular, our study of $Pr^\omega(L,l)$ addresses challenges that concern the periodic nature of lasso-shaped words and shows that the periodicity of lasso-shaped words affects their probability to be in L in ways that are orthogonal and not not less significant than their length.

2 Preliminaries

2.1 Automata

An *alphabet* Σ is a finite set of letters. A *word* over Σ is a finite or infinite sequence $w = \sigma_1, \sigma_2, \sigma_3, \cdots$ of letters from Σ. We use $|w|$ to denote the length of w, with $|w| = \infty$ for an infinite word w. We use Σ^* and Σ^ω to denote the set of all finite and infinite words over Σ, respectively. A *language* is a set of words. For a language $L \subseteq \Sigma^*$, we use $comp(L)$ to denote the language complementing L, thus $comp(L) = \Sigma^* \setminus L$.

A *nondeterministic automaton* is $\mathcal{A} = \langle \Sigma, Q, \delta, Q_0, \alpha \rangle$, where Σ is a finite input alphabet, Q is a finite set of states, $\delta : Q \times \Sigma \rightarrow 2^Q$ is a transition function, $Q_0 \subseteq Q$ is a set of initial states, and $\alpha \subseteq Q$ is a set of accepting states. We extend δ to finite words in the expected way. Thus, $\delta^*(q,u)$ is the state $\mathcal{A}$ reaches when it reads the word $u \in \Sigma^*$ from some state $q \in Q$. Formally, $\delta^* : Q \times \Sigma^* \rightarrow Q$ is such that for every $q \in Q$, we have that $\delta^*(q,\epsilon) = q$ and for a finite word $u \in \Sigma^*$ and letter $\sigma \in \Sigma$, we have that $\delta^*(q, u \cdot \sigma) = \delta(\delta^*(q,u), \sigma)$.

A *run* of $\mathcal{A}$ on a word w is the function $r : \{0 \leq i \leq |w|\} \to Q$, such that $r(0) \in Q_0$, and for all $i \geq 0$, we have that $r(i+1) \in \delta(r(i), \sigma_{i+1})$. If $|Q_0| = 1$, and for all $q \in Q$ and $\sigma \in \Sigma$, it holds that $|\delta(q, \sigma)| = 1$, then $\mathcal{A}$ has a single run on w, and we say that $\mathcal{A}$ is *deterministic*. We sometimes refer to a run also as a sequence of states; that is, $r = r(0), r(1), \ldots \in Q^{|w|+1}$.

When $\mathcal{A}$ runs on finite words, the run r is finite, and it is accepting iff it ends in an accepting state, thus $r(|w|) \in \alpha$. When $\mathcal{A}$ runs on infinite words, acceptance depends on the set $\inf(r)$, of the states that r visits infinitely often. Formally $\inf(r) = \{q \in Q :$ for infinitely many $i \in \mathbb{N}$, we have that $r(i) = q\}$. As Q is finite, the set $\inf(r)$ is guaranteed not to be empty. In *Büchi* automata, the run r is accepting iff r visits the set of accepting states infinitely often, thus $\inf(r) \cap \alpha \neq \emptyset$. Otherwise, r is rejecting. In *parity* automata, the acceptance condition is $\alpha : Q \to \{1, \ldots, k\}$, and a run r is accepting iff the minimal *color* in $\{1, \ldots, k\}$ that r visits infinitely often is even. The automaton $\mathcal{A}$ accepts a word w if there exists an accepting run r of $\mathcal{A}$ on w. The *language* of $\mathcal{A}$, denoted $L(\mathcal{A})$, is the set of words that $\mathcal{A}$ accepts. We also say that $\mathcal{A}$ recognizes $L(\mathcal{A})$. We define the size of $\mathcal{A}$, denoted $|\mathcal{A}|$, as the number of states that $\mathcal{A}$ has. A deterministic automaton $\mathcal{A}$ on finite words is *minimal* if there is no equivalent automaton, namely one that accepts the same language, of a smaller size.

We use NFW and DFW to denote nondeterministic and deterministic automata on finite words, respectively, and similarly for NBW, DBW, NPW, and DPW, denoting Büchi and parity automata.

An automaton $\mathcal{A} = \langle \Sigma, Q, q_0, \delta, \alpha \rangle$ is *weak* if there exists a partition of Q into sets $Q_1, Q_2, \ldots, Q_k$ such that for all $1 \leq i \leq k$, either $Q_i \subseteq \alpha$, in which case we say that Q_i is accepting, or $Q_i \cap \alpha = \emptyset$, in which case we say that Q_i is rejecting. In addition, there is a partial order $\leq$ on the sets such that transitions in δ lead to states of the same or of lower sets. Formally, for all states $q, q' \in Q$, if $q' \in \delta(q, \sigma)$, for some letter $\sigma \in \Sigma$, then the sets Q_i and Q_j for which $q \in Q_i$ and $q' \in Q_j$ satisfy $Q_j \leq Q_i$. Equivalently, the partition into strongly connected components of the graph induced by $\mathcal{A}$ is such that each component is either contained in α or disjoint from α. For $j \in \mathbb{N}$, we say that $\mathcal{A}$ is *weak*$[j]$ if $\mathcal{A}$ is weak and the largest set Q_i in the partition of Q is of size j. We refer to j as the *width* of $\mathcal{A}$. Describing classes of weak$[j]$ automata, we add $[j]$ to the acronym. For example an NFW[1] is a weak[1] NFW, namely an NFW in which all cycles are self loops. We sometimes refer also to the *depth* of $\mathcal{A}$, which is the maximal number of alternations between accepting and rejecting sets that a run may have.

A regular language L is *counter-free* (*CF*, for short) if there is $n \geq 1$ such that for every $m \geq n$ and $v, w, x \in \Sigma^*$, we have that $vw^nx \in L$ iff $vw^mx \in L$. For example, $L_1 = (ab)^*$ is CF, while $L_2 = (aa)^*$ is not CF. An NFW $\mathcal{A}$ is CF if $L(\mathcal{A})$ is CF. A DFW $\mathcal{A}$ is *permutation-free* (*PF*, for short) if there does not exist a non-trivial permutation between its states. That is, there does not exist a word $w \in \Sigma^*$ and a set $\{q_1, q_2, \ldots, q_l\}$, with $l \geq 2$ of different states in $\mathcal{A}$ such that $\delta^*(q_i, w) = q_{(i \mod l)+1}$ for all $1 \leq i \leq l$. A regular language is PF if its minimal automaton is PF. By [13] (Theorem 5.1), a regular language is CF iff its minimal DFW is PF.

A language $L \subseteq \Sigma^*$ is a *safety* language if every word not in L has a bad prefix. Formally, if $w \notin L$, then w has a prefix $x \in \Sigma^*$ such that for every $y \in \Sigma^*$, we have that $x \cdot y \notin L$. For example, if $\Sigma = \{a, b\}$, then the languages $L_1 = a^*$ is safety. Indeed, if $w \notin L_1$, then w has a prefix $x \in a^* \cdot b$, and $x \cdot y \notin L_1$ for all $y \in \Sigma^*$. On the other hand, the language $L_2 = (ab)^*$ is not safety, as, for example, the word aba is not in L_3, yet every prefix of it can be extended to a word in L_3. A language $L \subseteq \Sigma^*$ is a *co-safety* language if *comp*(L) is safety. Equivalently, every word $w \in L$ has a good prefix, namely a prefix x such that $x \cdot y \in L$ for all $y \in \Sigma^*$. Safety and co-safety languages can be recognized by weak automata of depth 1 [20, 12]. Indeed, in automata for safety languages, all runs start in accepting sets and move to a rejecting sink once a bad prefix is read. Dually, for co-safety languages, run start in rejecting sets and may move to an accepting sink.

The definitions of safety and co-safety languages apply also for languages $L \subseteq \Sigma^\omega$. Here, a bad prefix is $x \in \Sigma^*$ such that for every $y \in \Sigma^\omega$, we have that $x \cdot y \notin L$. Note that while the language $L_2 = (ab)^*$ of finite words is not safety, the language $L_2' = (ab)^\omega$ of infinite words is safety. Indeed, $w \notin L_2'$ iff w has a prefix ending with aa or bb, which is a bad prefix.

2.2 Monotonicity Characterizations of Regular Languages

Let $\{a_n\}_{n=1}^\infty = a_1, a_2, a_3, \ldots$ be some sequence of real numbers in $[0, 1]$. For convenience, we sometimes write $\{a_n\} = \{a_n\}_{n=1}^\infty$ for short. We say that the sequence $\{a_n\}_{n=1}^\infty$ is:

- *monotonically non-decreasing* (MND) if for all $n \geq 0$, we have that $a_{n+1} \geq a_n$. If for all $n \geq 0$, we have that $a_{n+1} > a_n$, then the sequence is *monotonically increasing* (MI).
- *monotonically non-increasing* (MNI) if for all $n \geq 0$, we have that $a_{n+1} \leq a_n$. If for all $n \geq 0$, we have that $a_{n+1} < a_n$, then the sequence is *monotonically decreasing* (MD).

We say that the sequence $\{a_n\}$ is *monotonic* (M) if $\{a_n\}$ is MNI or MND. Since we care about limit behavior, we have particular interest in sequences that are not immediately monotonic but rather monotonic from a certain index. For $\gamma \in \{\mathrm{MND}, \mathrm{MI}, \mathrm{MNI}, \mathrm{MD}, \mathrm{M}\}$, a sequence $\{a_n\}$ is *eventually* γ, if there exists $k \geq 0$, such that $\{a_n\}_{n=k}^\infty$ is γ. If $\{a_n\}$ is not EM, we say that it is *never-monotonic* (NM). If the sequence $\{a_n\}$ is not EM, we say that it is *never-monotonic* (NM). We refer to $\gamma \in \{$MND, MI ,MNI, MD, M, EMND, EMI, EMNI, EMD, EM, NM$\}$ as the *monotonicity characterization* of languages. We use $\tilde{\gamma}$ is the monotonicity characterization dual to γ. Thus, $\{a_n\}_{n=1}^\infty$ is γ iff $\{1 - a_n\}_{n=1}^\infty$ is $\tilde{\gamma}$. For example, $\widetilde{\mathrm{MI}} = \mathrm{MD}$.

Consider an alphabet Σ. We assume that letters in Σ are drawn uniformly at random (see Section 3.1 for an extension to arbitrary distributions). Accordingly, the probability of each letter to be drawn is $\frac{1}{|\Sigma|}$ and for a given length $n \geq 1$, the probability of a word of length n to be drawn is $\frac{1}{|\Sigma^n|}$. Consider a language $L \subseteq \Sigma^*$. The probability of a random word of length n to be in L is then $Pr(L, n) = \frac{|\{w \in \Sigma^n | w \in L\}|}{|\Sigma^n|}$. We characterize regular languages by the way the length of words influences membership in the language. For a language $L \subseteq \Sigma^*$ and a monotonicity characterization γ, we say that L is γ iff the sequence $\{Pr(L, n)\}_{n=1}^\infty$ is γ. Likewise, an automaton $\mathcal{A}$ is γ iff $L(\mathcal{A})$ is γ. Note that we start with $n = 1$ and ignore the membership of ϵ in L.

▶ **Example 1.** Consider the following languages over $\Sigma = \{a, b\}$.

- The language $L_1 = a^*$ induces the sequence $\{Pr(L_1, n) = \frac{1}{2^n}\}$, and is therefore MD.
- Its complement language $L_2 = \Sigma^* \cdot b \cdot \Sigma^*$ induces the sequence $\{Pr(L_2, n) = 1 - \frac{1}{2^n}\}$, and is therefore MI.
- Let $L_3 = (\epsilon + (\Sigma^* \cdot a)) \cdot (bb)^*$. Thus, L_3 contains exactly all words in which the number of b's after the last occurrence of a is even. The first elements in $\{Pr(L_3, n)\}$ are described in the table below

n	1	2	3	4
$Pr(L_3, n)$	$\frac{1}{2}$	$\frac{3}{4}$	$\frac{5}{8}$	$\frac{11}{16}$

For example, when $n = 3$, then out of 8 words of length 3, only the 5 words aaa, baa, bba, aba, and abb are in L_3. Although we can already determine that $\{Pr(L_3, n)\}$ is not M, it might be EM. ◀

▶ **Example 2.** Let $S_1, S_2 \subset \Sigma$ be a partition of an alphabet Σ into two nonempty sets. Consider the language $L_{once} = S_1^* \cdot S_2 \cdot S_1^*$. Thus, a word w is in L iff it contains exactly one occurrence of a letter in S_2. On the one hand, longer words are more likely to include a letter in S_2. On the other hand, longer words are more likely to include more than one such letter. Formally, if $p = \frac{|S_2|}{|\Sigma|}$ is the probability of a random letter to be in S_2, then it is not hard to show that $Pr(L_{once}, n) = n \cdot p \cdot (1-p)^{n-1}$. Thus, L_{once} is EM, yet monotonicity only starts when $n \geq \frac{1}{-\ln(1-p)}$. ⌟

3 Theoretical Properties

In this section we study some theoretical properties of monotonicity for regular languages. Recall that we define $Pr(L, n)$ as the probability of a word of length n to be in L, assuming that letters are drawn uniformly at random. In Section 3.1, we show that the probability according to which letters are drawn may actually change the monotonicity characteristic of languages, even in the limit. In Section 3.2, we study the sensitivity of the monotonicity to operations like complementation, union, and intersection. Finally, in Section 3.3, we consider the case of unary languages, thus when $\Sigma = \{a\}$ and $Pr(L, n) \in \{0, 1\}$ for all $n \geq 1$.

3.1 On the choice of a uniform distribution

Some properties, like a zero-one behavior of a language, are independent of the distribution of the letters in the alphabet [19]. In Theorem 3, we show that for monotonicity, the distribution may actually affect the characterization. Moreover, the distribution may not only turn monotonic languages into be eventually monotonic ones or turn strict monotonicity (that is, MI or MD) into non-strict one (that is, MND or MNI), but may turn a strictly monotonic language into one that is never monotonic:

▶ **Theorem 3.** *Let $\Sigma = \{a, b\}$. There is a language L such that L is MD when the letters in Σ are uniformly distributed and is NM in all distributions $f : \Sigma \to [0, 1]$ with $f^2(a) > f(b)$.*

Proof. Consider the language $L = aa(aa)^* + b(aa)^*$. Note that all the words in L are of the form $\sigma \cdot a^m$, with σ inducing the required parity of m: if $\sigma = a$, then m has to be odd, and if $\sigma = b$, then m has to even. Thus, for every length $n \geq 1$, there is exactly one word of length n in L. If n is even, then it is the word a^n, and if n is odd, then it is the word $b \cdot a^{n-1}$.

Accordingly, it is not hard to see that under a uniform distribution, we have that $Pr(L, n) = \frac{1}{2^n}$, and so L is MD. On the other hand, consider a distribution $f : \Sigma \to [0, 1]$ with $f(a) = p_a$ and $f(b) = p_b$. Then, $Pr_f(L, n) = p_a^n$ when n is even, and $Pr_f(L, n) = p_b \cdot p_a^{n-1}$ when n is odd. Note that when n is odd, then $Pr_f(L, n+1) = p_b \cdot Pr_f(L, n)$, thus $Pr_f(L, n+1) < Pr_f(L, n)$. In addition, $Pr_f(L, n+2) = p_a^2 \cdot Pr_f(L, n)$. Thus, if $p_a^2 > p_b$, then $Pr_f(L, n+2) > Pr_f(L, n+1)$. Thus, when $p_a^2 > p_b$, the sequence $\{Pr_f(L, n)\}_{n=1}^{\infty}$ is NM. For example, when $p_a = \frac{3}{4}$ and $f(b) = \frac{1}{4}$, we get that $Pr_f(L, n) = \frac{3^n(2+(-1)^n)}{4^n \cdot 3}$. ◀

Theorem 3 may question the robustness of our results. As we argue in Remark 20, however, all the results in the paper apply to arbitrary distributions.

3.2 Monotonicity characterization and Boolean operations

In this section we study whether the monotonicity characterization of languages is preserved under complementation, union, and intersection.

We start with complementation. Since for every language L, and for every $n \geq 1$, we have that $Pr(comp(L), n) = 1 - Pr(L, n)$, dualization follows immediately from the definitions:

▶ **Theorem 4.** *For every language $L \subseteq \Sigma^*$ and monotonicity characterization γ, we have that L is γ iff $comp(L)$ is $\tilde{\gamma}$.*

We continue to union and intersection. It is easy to come up with languages with dual monotonicity characterization whose union and intersections are not monotonic. We show that, surprisingly, also the union and intersection of languages that are both monotonically increasing or both monotonically decreasing, need not be monotonic, even in the limit.

▶ **Theorem 5.** *The intersection and union of MD (or MI) languages may be NM.*

Proof. We prove the result for MD languages. By considering the complement languages, one can get proofs for MI languages.

We start with intersection. Consider the languages $L_1 = a^*$ and $L_2 = (aa)^* + b(bb)^*$. It is not hard to show that for every $n \geq 0$, we have that $Pr(L_1, n) = Pr(L_2, n) = \frac{1}{2^n}$. Thus, both languages are MD. Let $L_\cap = L_1 \cap L_2$. Note that $L_\cap = (aa)^*$. Accordingly, $Pr(L_\cap, n)$ is 0 for odd n's and is $\frac{1}{2^n}$ for even n's. Hence, $L_\cap$ is NM.

We continue with union. Consider the languages $L_3 = a\Sigma^* + b^*$ and $L_4 = a\Sigma(\Sigma\Sigma)^* + (bb)^* + b(\Sigma\Sigma)^* + a(aa)^*$. It is not hard to see that for every $n \geq 1$, we have that, $Pr(L_3, n) = Pr(L_4, n) = \frac{1}{2} + \frac{1}{2^n}$. Thus, both languages are MD. Let $L_\cup = L_3 \cup L_4$. Note that $L_\cup = a\Sigma^* + (bb)^* + \Sigma(\Sigma\Sigma)^*$. Thus, $Pr(L_\cup, n)$ is 1 for odd n's and is $\frac{1}{2} + \frac{1}{2^n}$ for even n's. Hence, $L_\cup$ is NM. ◀

3.3 The case of unary languages

In this section we examine languages over a unary alphabet, thus when $\Sigma = \{a\}$. Note that then, each language $L \subseteq a^*$ corresponds to a set of integers, namely all $n \geq 0$ such that $a^n \in L$. Accordingly, $Pr(L, n) = 1$ if $a^n \in L$, and $Pr(L, n) = 0$ if $a^n \notin L$. It follows that a unary language cannot be MI or MD, and that MND and MNI unary languages are trivial. Indeed, only $L = a^*$ and $L = \emptyset$ have $Pr(L, n) = 1$ and $Pr(L, n) = 0$ for all $n \geq 0$, respectively. Thus, a unary language $L \subseteq a^*$ can have one of the following two monotonicity characteristics.

- EM, which holds if L is *almost trivial*, thus L or $comp(L)$ are finite. Formally, there is $m \geq 0$ such that either $a^k \in L$ for all $k \geq m$, or $a^k \notin L$ for all $k \geq m$.
- NM if for all $k \geq 0$, we have $m, l \geq k$ with $a^m \in L$ and $a^l \notin L$.

Note that the definitions coincide with these presented in Section 2.2, simplified to the case $Pr(L, n)$ is in $\{0, 1\}$ for all $n \geq 1$.

▶ **Theorem 6.** *Unary NFW[1]s are EM.*

Proof. Consider an NFW[1] $\mathcal{A} = \langle \{a\}, Q, \delta, Q_0, \alpha \rangle$ with m states. We distinguish between two cases. First, if $\mathcal{A}$ has a state $q \in \alpha$ that is reachable from Q_0 by a path that includes a state with a self loop (possibly q itself has a self loop), then $\{a^k : k \geq m\} \subseteq L(\mathcal{A})$, and so $\{Pr(L(\mathcal{A}), n)\}_{n=1}^{\infty}$ is eventually always 1. Otherwise, all the states in α are reachable only by paths that do not include states with self loops. Thus, $|L(\mathcal{A})| \leq |\alpha|$, and so $\{Pr(L(\mathcal{A}), n)\}_{n=1}^{\infty}$ is eventually always 0. ◀

Note that the graph induced by a unary DFW $\mathcal{A}$ that accepts an infinite language must be *lasso-shaped*. Also, $\mathcal{A}$ is EM iff all the states in the cycle of the lasso are accepting or all are not accepting. Accordingly, arbitrary unary DFWs may be NM, yet all weak unary DFWs are EM. In the nondeterministic setting, the characterization in Theorem 6 is tight, and even unary NFW[2]s may be NM:

▶ **Theorem 7.** *Unary NFW[2]s may be NM.*

Proof. Consider the NFW[2] appearing in Figure 1. It is not hard to see that its language is $(aa)^+$, which is NM. ◀

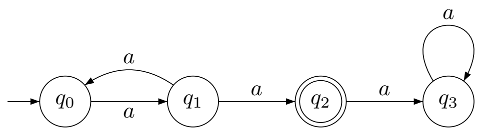


Figure 1 A unary NFW[2] whose language is $(aa)^+$.

4 An Algebraic Approach

In this section we relate monotonicity characterization with properties of the stochastic matrix that describes the behavior of a given DFW. We first need some definitions.

A *Markov chain* is a tuple $M_\mathcal{A} = \langle Q, \tau \rangle$, where Q is a set of states and $\tau : Q \times Q \to [0,1]$ is a probabilistic transition function: for $q, q' \in Q$, the value $\tau(q, q')$ is the probability of moving from q to q'. Since τ describes a distribution, then for every $q \in Q$, we have that $\sum_{s \in Q} \tau(q, s) = 1$.

Each DFW $\mathcal{A} = \langle \Sigma, Q, \delta, q_0, F \rangle$ induces a Markov chain $M_\mathcal{A} = \langle Q, \tau \rangle$, where for $q, q' \in Q$, we have that $\tau(q, q')$ is the probability of a run that visits state q to move to state q' when it reads the next letter. Thus, assuming a uniform distribution on Σ, we have that $\tau(q, q') = \frac{|\{\sigma : \delta(q,\sigma) = q'\}|}{|\Sigma|}$.

▶ **Example 8.** Consider the DFW $\mathcal{A}$ appearing in Figure 2. Its induced Markov chain $M_\mathcal{A}$ appears to its right.

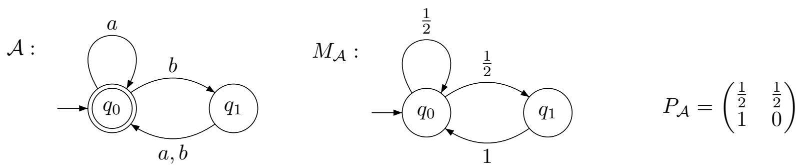


Figure 2 A DFW $\mathcal{A}$, its induced Markov chain $M_\mathcal{A}$, and its stochastic matrix $P_\mathcal{A}$.

Note that $\mathcal{A}$ recognizes the language L_3 discussed in Example 1, namely the language of all words that have an even number of b's after the last occurrence of a. In Example 1, we described the first elements in $\{Pr(L_3, n)\}$. In particular, we calculated $Pr(L_3, 3)$ by counting the number of words of length 3 that are in L_3. We now show how to calculate $Pr(L_3, 3)$ by examining the Markov chain $M_\mathcal{A}$. A word of length 3 is accepted by $\mathcal{A}$ only if its run on $\mathcal{A}$ is $q_0q_0q_0q_0$, which happens with probability $\frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2}$, or $q_0q_1q_0q_0$, which happens with probability $\frac{1}{2} \cdot 1 \cdot \frac{1}{2}$, or $q_0q_0q_1q_0$, which happens with probability $\frac{1}{2} \cdot \frac{1}{2} \cdot 1$. Overall, we have that $Pr(L(\mathcal{A}), 3) = \frac{5}{8}$. ⌟

The *stochastic matrix* of a DFW $\mathcal{A}$, denoted $P_\mathcal{A}$, describes the transition function τ of its Markov chain $M_\mathcal{A}$. Formally, we assume some order on the states in Q, thus $Q = \{q_1, q_2, \ldots, q_n\}$, and $P_\mathcal{A}$ is an $n \times n$ matrix with elements in $[0,1]$, where for $1 \leq i, j \leq m$ we have that $P_{\mathcal{A}_{i,j}} = \tau(q_i, q_j)$. Given a DFW $\mathcal{A}$ and states $q_i, q_j \in Q$, the probability of a run on a word of length n that starts in q_i to end in the state q_j is $(P^n_\mathcal{A})_{i,j}$. Let x^n_i denote the probability of a word of length n to end in state q_i. If the initial state of $\mathcal{A}$ is q_1, then $x^n_i = (P^m_\mathcal{A})_{1,i}$.

▶ **Example 9.** Back to the DFW $\mathcal{A}$ from Figure 2. The matrix $P_{\mathcal{A}}$ appears on the right of the figure. Recall that x_0^n and x_0^n denote the probability of a word to end on the states q_0 and q_1, respectively. By diagonalizing $P_{\mathcal{A}}$ (see details in Appendix A), we can calculate:

$$(x_0^n, x_1^n) = (1,0) \cdot P_{\mathcal{A}}^n = \left(\frac{2 + (\frac{-1}{2})^n}{3}, \frac{1 - (\frac{-1}{2})^n}{3} \right).$$

Since $\alpha = \{q_0\}$, we have that $Pr(L(\mathcal{A}), n) = x_0^n = \frac{2+(\frac{-1}{2})^n}{3}$. Note that $Pr(L(\mathcal{A}), n) > Pr(L(\mathcal{A}), n-1)$, when n is even, and $Pr(L(\mathcal{A}), n) < Pr(L(\mathcal{A}), n-1)$, when n is odd. Hence, $L(\mathcal{A})$ is NM. ⌟

An *eigenvalue* of matrix A is a scalar, denoted λ_i, such that there exists a non-zero vector v for which $Av = \lambda_i v$. In Appendix A, we describe how to reason about $(P_{\mathcal{A}}^m)_{1,i}$ using known results from linear algebra. Specifically, we prove the following lemma.

▶ **Lemma 10.** *Consider a stochastic matrix $P \in \mathbb{R}^{d \times d}$ and a set $S = \{(i_1, j_1), \cdots, (i_t, j_t)\}$ of t pairs of indices in P. If all the eigenvalues of P are real and non-negative, then the sequence $\{\sum_{(i',j') \in S} P^n_{i',j'}\}_{n=1}^{\infty}$ is EM.*

Lemma 10 is useful for reasoning about the monotonicity of DFWs.

▶ **Theorem 11.** *Let $\mathcal{A}$ be some DFW. If $P_{\mathcal{A}}$ has only real and non-negative eigenvalues, then $L(\mathcal{A})$ is EM.*

Proof. Let $q_{l_1}, q_{l_2}, \ldots, q_{l_t}$ be the accepting states of $\mathcal{A}$, and let q_s be its initial state. Let $S = \{(s, l_1), (s, l_2), \ldots, (s, l_t)\}$. As $\mathcal{A}$ is deterministic, then $Pr(L(\mathcal{A}), n) = \sum_{(i',j') \in S} P^n_{i',j'}$. By Lemma 10, we thus have that $\{Pr(L, n)\}$ is EM. ◀

5 Classes of Monotonic Languages

In this section we point to three classes of monotonic languages. The first two refer to weak automata and the third to permutation-free automata. The classes are tight, in the sense that if we remove some limitations in their definition, we end up in classes that are never monotonic.

5.1 Weak automata – width

We start with the width of weak automata and show that if the sets in the partition that witnesses the weakness of the automaton are singletons, then its language is EM, and that this sufficient condition is tight.

▶ **Theorem 12.** *DFW[1]s are EM.*

Proof. Let $\mathcal{A}$ be some DFW[1] and let $\{q_0\} \leq \{q_1\} \leq \ldots \leq \{q_{m-1}\}$ be the partition of the states of $\mathcal{A}$ into sets that witness its weakness. Then, for every $i < j$, we have that $\tau(q_i, q_j) = 0$, implying that $P_{\mathcal{A}}$ is an upper triangular matrix, and so its eigenvalues are on its diagonal. Hence, as the elements of $P_{\mathcal{A}}$ are real and non-negative, so are it eigenvalues. Thus, by Theorem 11, we have that $\mathcal{A}$ is EM. ◀

▶ **Example 13.** Recall the language L_{once} from Example 2. Figure 3 describes a DFW[1] for L_{once}, which immediately implies it is EM. ⌟

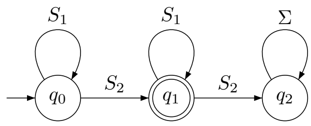


Figure 3 A DFW[1] for L_{once}.

We now show that Theorem 12 is tight, in the sense we cannot remove any of the limitations imposed by the DFW[1] restrictions. Thus, DFW[2]s and NFW[1]s need not be monotonic. Note that this is in contrast with the unary case, where NFW[1]s are EM (Theorem 6).

▶ **Theorem 14.** *DFW[2]s may be NM.*

Proof. Consider the DFW[2] $\mathcal{D}$ that appears in Figure 4 (left). It is not hard to prove that $L(\mathcal{D}) = a(ba)^*a$. Thus, $\mathcal{D}$ only accept words of even length. Moreover, while the probability of words of odd length to be accepted is 0, the probability is positive for words of even length. Specifically, $Pr(L(\mathcal{D}), n) = \frac{(-1)^n+1}{2^{n+1}}$, which is NM. ◀

▶ **Theorem 15.** *NFW[1]s may be NM.*

Proof. Let $\Sigma = \{a, b\}$ and consider the NFW $\mathcal{A}$ appearing in Figure 4 (right). It is not hard to see that $L(\mathcal{A}) = comp((ab)^*)$. Indeed, $comp((ab)^*) = \Sigma^* \cdot a + b \cdot \Sigma^* + \Sigma^* \cdot (aa + bb) \cdot \Sigma^*$. Since $(ab)^*$ is infinite and contains only words of even length, it is NM. Hence, by Theorem 4, so is $L(\mathcal{A})$. ◀

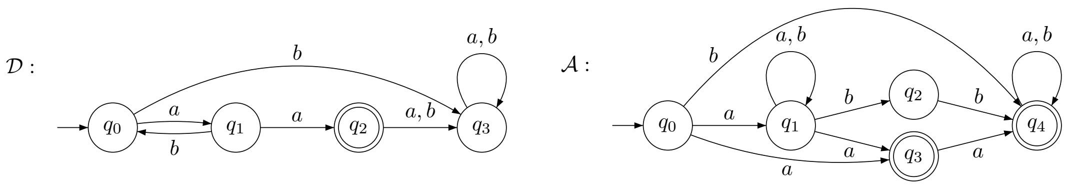


Figure 4 A DFW[2] recognizing $a(ba)^*a$ and an NFW[1] recognizing $comp((ab)^*)$.

5.2 Weak automata – depth

We continue to the depth of weak automata, namely the bound on the number of alternations between accepting and rejecting sets in the partition that witnesses the weakness. Weak automata of depth 1 recognize safety and co-safety languages [20, 12]. Intuitively, the longer a word is, the more likely it is for a bad thing to do happen. Formally, we have the following.

▶ **Theorem 16.** *All safety (co-safety) languages of finite words are MNI (MND, respectively).*

Proof. We prove that all co-safety languages are MND. The result for safety follows from Theorem 4. For each $n \geq 0$, we calculate the number $\#(L, n)$ of words in $L \cap \Sigma^n$. First $\#(L, 0) \in \{0, 1\}$, according to the membership of ϵ in L. Then, counting words in $L \cap \Sigma^{n+1}$, observe that since L is co-safety, then each word in $L \cap \Sigma^n$ contributes to $L \cap \Sigma^{n+1}$ all its $|\Sigma|$ extensions by one letter. Thus, $\#(L, n+1) \geq \#(L, n) \cdot |\Sigma|$. Since $\#(L, n) = Pr(L, n) \cdot |\Sigma|^n$, we get that

$$Pr(L, n+1) \geq \frac{Pr(L, n) \cdot |\Sigma|^n \cdot |\Sigma|}{|\Sigma|^{n+1}} = Pr(L, n).$$ ◀

Recall the DFW[2] $\mathcal{D}$ and NFW[1] $\mathcal{A}$ that are described in Figure 4. In Theorem 14, we proved that their languages are NM. As $\mathcal{D}$ has depth 2 and $\mathcal{A}$ has depth 1, they also serve for proving the tightness of Theorem 16. Thus, we have the following.

▶ **Theorem 17.** *All weak DFWs of depth* 1 *are M. On the other hand, there is a weak DFW of depth* 2 *and a weak NFW of depth* 1 *whose languages are NM.*

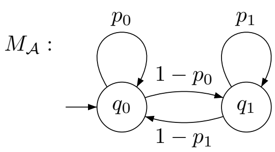
5.3 Permutation-free automata

We conclude with automata that recognize CF languages, and give a tight sufficient condition also for them.

▶ **Theorem 18.** *2-state CF DFWs are M.*

Proof. In Figure 5, we describe a general 2-state DFW $\mathcal{A} = \langle \Sigma, \{q_0, q_1\}, \delta, q_0, \alpha \rangle$, with its stochastic matrix $P_\mathcal{A}$. If both q_0, q_1 are accepting or both are rejecting, then $Pr(L(\mathcal{A}), n) = 1$ or 0 respectively. We assume that exactly one of the states of $\mathcal{A}$ is accepting. Let $\tau(q_0, q_0) = p_0$ and $\tau(q_1, q_1) = p_1$, where τ is the transition function of the Markov chain $M_\mathcal{A}$. Consider the stochastic matrix $P_\mathcal{A}$ of $\mathcal{A}$, shown in Figure 5. The eigenvalues of $P_\mathcal{A}$ can be calculated and are $\lambda_1 = 1$ and $\lambda_2 = p_0 + p_1 - 1$ (note that λ_2 might be equal to 1 as well). Note that $Pr(L(\mathcal{A}), n) = c_1 + c_2(p_0 + p_1 - 1)^n$, for some $c_1, c_2 \in \mathbb{R}$. We show that $p_0 + p_1 - 1 \geq 0$, implying that $\mathcal{A}$ is M.

Assume by way of contradiction that $p_0 + p_1 - 1 < 0$. Then $(1 - p_0) + (1 - p_1) > 1$. Therefore, there exists some $\sigma \in \Sigma$, such that $\delta(q_0, \sigma) = q_1$ and $\delta(q_1, \sigma) = q_0$, and so there is a non-trivial permutation in $\mathcal{A}$. Since only one of the states of $\mathcal{A}$ is accepting, it is minimal. Thus, $\mathcal{A}$ is PF, and we have reached a contradiction. ◀



$$P_\mathcal{A} = \begin{pmatrix} p_0 & 1 - p_0 \\ 1 - p_1 & p_1 \end{pmatrix}$$

Figure 5 A CF DFW $\mathcal{A}$ with two states.

▶ **Theorem 19.** *2-state DFWs, 3-state CF DFWs, and 2-state CF NFWs may be NM.*

Proof. First, in Example 9, we saw that the 2-state DFW from Example 8 is NM. For a 3-state CF DFW, consider the DFW $\mathcal{D}$ appearing in Figure 6. It is not hard to prove that $\mathcal{D}$ is PF and that $L(\mathcal{D}) = comp((ab)^*)$, which is NM. Now, for a 2-state CF NFW, consider automaton $\mathcal{A}$ appearing to the right of $\mathcal{D}$. Note that $L(\mathcal{A}) = L(\mathcal{D})$, Indeed, $\mathcal{D}$ can be obtained by applying the subset construction on $\mathcal{A}$. Thus, $\mathcal{A}$ is a 2-state CF NFW whose language is NM. ◀

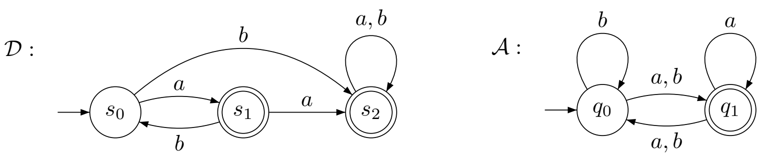


Figure 6 A 3-state DFW and a 2-state NFW for the PF and NM language $comp((ab)^*)$.

▶ Remark 20. It is not hard to check that all the positive results in this section can be extended to nonuniform distribution of the letters in the alphabet. Indeed, the proofs of Theorems 12 and 18 are independent of the uniform distributions, and the proof of Theorem 16 can be easily adjusted to the general case. ⌟

▶ Remark 21. The logic LTL_f is a linear temporal logic that specifies languages of finite words. By [6], PF DFWs are as expressive as LTL_f. In particular, the LTL_f formula $p \wedge ((p \wedge X\neg p) \vee (\neg p \wedge Xp))U(p \wedge X(p \wedge \mathit{last}))$ is satisfied only by computations of even length, and is similar to the NM DFW[2] from Figure 4. ⌟

6 Infinite Words

For languages of infinite words, we consider lasso-shaped words, thus words of the form vu^ω, for some $v, u \in \Sigma^*$. We study three sequences. The first two refer to the prefix of the lasso, namely to the probability of prefixes $v \in \Sigma^n$ to be extendable, by some or all suffixes, to words in L. The third refers to the loop of the lasso, namely to the probability of a loop $u \in \Sigma^n$ to be such that $u^\omega \in L$. We relate the three sequences to sequences induced by languages of finite words. While this is straightforward for the two sequences that refer to the prefix, it involves new ideas and constructions for the third sequence, where reasoning depends not only on the length of the lasso, but also on the words being periodic.

6.1 Monotonicity in the length of the prefix

For a language $L \subseteq \Sigma^\omega$, let $\mathit{good.pref}^\exists(L) = \{v \in \Sigma^* \mid \text{there is } u \in \Sigma^\omega \text{ such that } v \cdot u \in L\}$ and $\mathit{good.pref}^\forall(L) = \{v \in \Sigma^* \mid \text{ for all } u \in \Sigma^\omega \text{ we have that } v \cdot u \in L\}$. We define $Pr^\exists(L, n) = \frac{|\mathit{good.pref}^\exists(L) \cap \Sigma^n|}{|\Sigma^n|}$ and $Pr^\forall(L, n) = \frac{|\mathit{good.pref}^\forall(L) \cap \Sigma^n|}{|\Sigma^n|}$. Thus, $\{Pr^\exists(L, n)\}_{n=1}^\infty$ and $\{Pr^\forall(L, n)\}_{n=1}^\infty$ describe how the length of words influences their probability to be extendable to words in L by some or by all suffixes, respectively. Intuitively, the longer a prefix is, the smaller is the probability to extend it to a word in L by some suffix, yet the higher is the probability that all its extensions result in a word in L. Formally, we have the following.

▶ **Theorem 22.** *For every language $L \subseteq \Sigma^\omega$, we have that $\{Pr^\exists(L, n)\}_{n=1}^\infty$ is MNI and $\{Pr^\forall(L, n)\}_{n=1}^\infty$ is MND.*

Proof. It is easy to see that $\mathit{good.pref}^\exists(L)$ is a safety language. Indeed, if $v \in \Sigma^*$ cannot be extended to a word in L, then so do all its extensions. Also, as $\mathit{good.pref}^\forall(L) = \mathit{comp}(\mathit{good.pref}^\exists(\mathit{comp}(L)))$, we have that $\mathit{good.pref}^\forall(L)$ is a co-safety language.

By definition, $Pr^\exists(L, n) = Pr(\mathit{good.pref}^\exists(L), n)$ and $Pr^\exists(L, n) = Pr(\mathit{good.pref}^\forall(L), n)$. Thus, the claim follows from Theorem 16. ◀

6.2 Monotonicity in the length of the loop

We continue and study the sequence$\{Pr^\omega(L, n)\}_{n=1}^\infty$. Recall that $Pr^\omega(L, n)$ is the probability of a word $u \in \Sigma^n$ to be such that $u^\omega \in L$. Formally, $Pr^\omega(L, n) = \frac{|\{u \in \Sigma^n | u^\omega \in L\}|}{|\Sigma^n|}$. Thus, here, in addition to the dependency of membership in L in the length of u, monotonicity may depend on the periodic nature of words of the form u^ω.

In order to illustrate this dependency, we start with a simple observation.

▶ **Theorem 23.** *Safety and co-safety languages of infinite words may be NM.*

Proof. Consider the safety language $L = (ab)^\omega$. Note that for all $n \geq 0$, we have that $Pr^\omega(L, 2n) > 0$, whereas $Pr^\omega(L, 2n+1) = 0$. Indeed, for $u = (ab)^n$, we have that $u^\omega \in L$, whereas for u of an odd length, either u contains aa or bb, or the first and last letters of u are identical. In both cases, $u^\omega \notin L$. Dually, $comp(L)$ is a co-safety language, and for all $n \geq 0$ we have that $Pr^\omega(L, 2n+1) = 1$, whereas $Pr^\omega(L, 2n) < 1$. ◀

On the other hand, it is not hard to extend the results in Section 3.2 about complementation, union, and intersection, also to languages on infinite words.

▶ **Theorem 24.** *For every language $L \subseteq \Sigma^\omega$ and monotonicity characterization γ, we have that L is γ iff $comp(L)$ is $\tilde{\gamma}$. On the other hand, the intersection and union of MD (or MI) languages of infinite words may be NM.*

Proof. The proof for complementation is identical to the one in the case of finite words. Indeed, also in the case of infinite words, we have that $Pr^\omega(comp(L), n) = \frac{|\{u \in \Sigma^n | u^\omega \notin L\}|}{|\Sigma^n|} = 1 - \frac{|\{u \in \Sigma^n | u^\omega \in L\}|}{|\Sigma^n|} = 1 - Pr^\omega(L, n)$.

For union and intersection, we prove the result for MD languages over the alphabet $\Sigma = \{a, b, c, d\}$. We start with intersection. Consider the languages $L_1 = a^\omega + (ab)^\omega$ and $L_2 = b^\omega + (ab)^\omega$. Observe that a finite word $w \in \Sigma^*$ is such that $w^\omega \in L_1$ iff $w = a^*$ or $w = (ab)^*$. Therefore, $Pr(L_1, n)$ is $\frac{1}{4^n}$ for odd n's, and is $\frac{2}{4^n}$ for even n's. Since for every $n \geq 1$, we have that $\frac{2}{4^n} > \frac{1}{4^{n+1}}$ and $\frac{1}{4^n} > \frac{2}{4^{n+1}}$, it follows that L_1 is MD. A similar argument implies that L_2 is MD. Let $L_\cap = L_1 \cap L_2$. Note that $L_\cap = (ab)^\omega$. Thus, $w^\omega \in L_\cap$ iff $w = (ab)^*$, and so $L_\cap$ is NM.

For union, consider the MD languages $L_3 = a^\omega + (ab)^\omega + (ac)^\omega$ and $L_4 = a^\omega + (ba)^\omega + (bc)^\omega$. Let $L_\cup = L_3 \cup L_4$. Note that $L_\cup = a^\omega + (ab)^\omega + (ac)^\omega + (ba)^\omega + (bc)^\omega$. It is not hard to show that $Pr(L_\cup, n)$ is $\frac{1}{4^n}$ for odd n's, and is $\frac{5}{4^n}$ for even n's. Since $Pr(L(\mathcal{A}), n) > Pr(L(\mathcal{A}), n+1)$ for odd n's, and $Pr(L(\mathcal{A}), n) < Pr(L(\mathcal{A}), n+1)$ for even n's, we have that $L_\cup$ is NM. ◀

We continue and examine CF languages of infinite words. Recall that for finite words, we showed that every 2-state CF DFW is EM, and that the characterization is tight. By going over all 2-state NBWs, one can show that they are all M. In fact, the latter holds regardless of the distribution on the letters. On the other hand, the 3-state DBW for $comp((ab)^\omega)$ is CF and NM. Hence, we have the following.

▶ **Theorem 25.** *2-states NBWs are M, yet 3-state CF NBWs may be NM.*

Proof. Consider the 2-state NBW $\mathcal{A} = \langle \Sigma, \{q_0, q_1\}, q_0, \delta, \alpha \rangle$. If both states are accepting or rejecting, then $Pr(L(\mathcal{A}), n) = 1$ or 0 respectively. Let $\Sigma_{i \to j} = \{\sigma \in \Sigma \mid \delta(q_i, \sigma) = q_j\}$. If $\alpha = \{q_0\}$, then a finite word $w \in \Sigma^*$ is such that $w^\omega \notin L(\mathcal{A})$ if every run of w reaches the state q_1 and there does not exist a letter σ in w such that $\sigma \in \Sigma_{1 \to 0}$. Let f be some distribution on Σ. The probability that all the runs on w^ω, for a random word w of length n to be rejecting, under a distribution f, is exactly $(\sum_{\sigma \in \Sigma \setminus \Sigma_{1 \to 0}} f(\sigma))^{n-1} \cdot (\sum_{\sigma \in \Sigma_{0 \to 1} \cap \Sigma \setminus (\Sigma_{1 \to 0} \cup \Sigma_{0 \to 0})} f(\sigma))$. Since $\sum_{\sigma \in \Sigma \setminus \Sigma_{1 \to 0}} f(\sigma) \geq 0$, we have that $\mathcal{A}$ is M.

If $\alpha = \{q_1\}$, then a finite word $w \in \Sigma^*$ is such that $w^\omega \notin L(\mathcal{A})$ if every run of w never visits q_1. The probability that random word w of length n to be such that $w^\omega \notin L(\mathcal{A})$, under a distribution f, is then $(\sum_{\sigma \in \Sigma \setminus \Sigma_{0 \to 1}} f(\sigma))^n$. Hence, $\mathcal{A}$ is M. ◀

Note that all LTL formulas induce CF languages [21]. For example, the infinite counterpart of the language from Example 2 is $L^\omega_{once} = S_1^* \cdot S_2 \cdot S_1^\omega$, which corresponds to the formula $S_1 U (S_2 \wedge XG(S_1))$. Note that for all $n \geq 1$, we have that $Pr^\omega(L^\omega_{once}, n) = 0$, as a periodic word includes either none or infinitely many occurrences of S_2. On the other hand, for the language L induced by the formula $GF(S_2)$, we have $Pr^\omega(L,n) = 1 - \frac{1}{2^n}$, which is strictly monotonic. Indeed, L can be recognized by a 2-state NBW. A more surprising example is $L' = GF(S_2 \wedge X(S_2))$, which requires infinitely many successive occurrences of S_2. Note that the length of the required sequence of S_2's is even, and so the periodic nature of words may hint that L' is not monotonic. However, as L' can be recognized by a 2-state NBW, Theorem 25 implies that it is actually monotonic.

We continue to our most technically elaborated result. For $L \subseteq \Sigma^\omega$, let $L^{\frac{1}{\omega}} = \{u \in \Sigma^* : u^\omega \in L\}$. We describe a construction that enables us to reduce reasoning about $Pr^\omega(L,n)$ to reasoning about $Pr(L^{\frac{1}{\omega}}, n)$. In Theorem 26 we describe the construction for L given by a DPW, which involves an exponential blow-up. Then, in Theorem 27, we prove an exponential lower bound for the construction, in fact for a family of languages definable by DBWs of depth 1.

▶ **Theorem 26.** *For every DPW $\mathcal{A}$, there exists a DFW $\mathcal{A}^{\frac{1}{\omega}}$ such that $L(\mathcal{A}^{\frac{1}{\omega}}) = L(\mathcal{A})^{\frac{1}{\omega}}$. The construction of $\mathcal{A}^{\frac{1}{\omega}}$ is effective and its size is exponential in the size of $\mathcal{A}$.*

Proof. Let $\mathcal{A} = \langle \Sigma, Q, \delta, q_0, \alpha \rangle$, with $\alpha : Q \to \{1, \dots, k\}$. For two functions $s : Q \to Q$ and $c : Q \to \{1, \dots, k\}$, and a word $u \in \Sigma^*$, we say that the pair $\langle s, c \rangle$ *captures* u if for all states $q \in Q$, we have that $s(q) = \delta^*(q, u)$, and $c(q) = \min\{\alpha(\delta^*(q, v)) : v \text{ is a prefix of } u\}$. Thus, s describes how each of the states in Q proceeds when it reads u, and c describes the minimum color that is visited along the way. Note that since $\mathcal{A}$ is deterministic, then every word u has a single pair of functions that captures it. We combine s and c to a single function $f : Q \to Q \times [k]$, and define successive applications of f by $f^0(q) = \langle q, \alpha(q) \rangle$ and $f^{i+1}(q) = \langle \delta^*(s^i(q), u), \min\{c^i(q), c(s^i(q)\}\rangle$. Thus, $f^i(q) = \langle q', j \rangle$, for $q' = \delta^*(q, u^i)$ and j being the minimum color that is visited when u^i is read from q.

Let $f : Q \to Q \times [k]$ be a function that captures a word $u \in \Sigma^*$, and let $f = \langle s, c \rangle$. By the pigeonhole principle, there must be $0 \le i_1 < i_2 \le |Q|$ for which $s^{i_1}(q_0) = s^{i_2}(q_0)$. Let $i_1 < i_2$ be such indices that minimize i_2, and let q be the state such that $s^{i_1}(q_0) = s^{i_2}(q_0) = q$. By the definition of acceptance in parity automata, we have that u^ω is accepted from q_0 iff $c^{i_2 - i_1}(q)$ is even. Indeed, the run from q_0 on u^ω starts by reaching q after reading u^{i_1}, and then repeatedly traverses a loop in q while reading $u^{i_2 - i_1}$, with $c^{i_2 - i_1}(q)$ being the minimal color that is visited along the loop, making it the minimal color visited infinitely often while reading u^ω. Accordingly, we say that a function $f : Q \to Q \times [k]$ is *good* if the minimal indices $0 \le i_1 < i_2$ for which $s^{i_1}(q_0) = s^{i_2}(q_0)$ are such that $c^{i_2 - i_1}(s^{i_1}(q_0))$ is even. Note that the definition of f is independent of a word. Yet, a word $u \in \Sigma^\omega$ is accepted by $\mathcal{A}$ iff the function f that captures it is good. Also note that deciding whether a given function f is good can be done in polynomial time.

We define the DFA $\mathcal{A}^{\frac{1}{\omega}} = \langle \Sigma, \mathcal{F}, \rho, f_0, \alpha' \rangle$, where $\mathcal{F}$ is the set of functions $f : Q \to Q \times \{1, \dots, k\}$, and f_0 and ρ are defined so that after reading a word $u \in \Sigma^*$, the DFA reaches a state that captures u. Formally, the initial state f_0 is the function that captures ϵ, thus $s_0(q) = q$ and $c_0(q) = \alpha(q)$, for all $q \in Q$, and the transitions function ρ is such that for every state $\langle s, c \rangle \in \mathcal{F}$ and letter $\sigma \in \Sigma$, we have that $\rho(\langle s, c \rangle, \sigma) = \langle s', c' \rangle$, where for every $q \in Q$, we have that $s'(q) = \delta(s(q), \sigma)$ and $c'(q) = \min\{c(q), \alpha(s'(q))\}$. Finally, α' is the set of all good functions. Since a word $u \in \Sigma^\omega$ is accepted by $\mathcal{A}$ iff the function f that captures it is good, we have that $L(\mathcal{A}^{\frac{1}{\omega}}) = L(\mathcal{A})^{\frac{1}{\omega}}$, and we are done. ◀

▶ **Theorem 27.** *There is a family $L_1, L_2, L_3, \dots$ of languages of infinite words such that for every $n \ge 1$, there is a DBW of depth 1 with $O(n)$ states that recognizes L_n, yet every DFW that recognizes $L_n^{\frac{1}{\omega}}$ needs at least 2^n states.*

Proof. For $n \geq 1$, let $\Sigma_n = \{1, 2, \ldots, n\}$. We define

$$L_n = \{\Sigma_n^* \cdot \# \cdot \sigma_1 \cdot \sigma_2 \cdots \sigma_k \cdot \# \cdot (\Sigma_n + \#)^\omega : \sigma_1 \in \{\sigma_2, \ldots, \sigma_k\}\}.$$

Note that $L_n \subseteq (\Sigma_n \cup \{\#\})^\omega$. First, a DBW for L_n waits for the first $\#$, records in its state space the letter σ after it, then waits for σ to appear before a second $\#$, after which the DBW moves to an accepting sink. If $\sigma = \#$ or if the second $\#$ appears before σ reappears, the DBW goes to a rejecting sink. Clearly, the DBW needs only $O(n)$ states. Also, each word in L_n has a good prefix, and so L_n is a co-safety language, and the DBW is of depth 1.

Assume by way of contradiction that there is a DFW $\mathcal{A}_n = \langle \Sigma_n, Q, q_0, \delta, \alpha \rangle$ that recognizes $L_n^{\frac{1}{\omega}}$ and has less than 2^n states. For a set $S \subseteq \Sigma_n$, let $w_S \in \Sigma_n^*$ be a word that contains exactly all the letters in S. Since $\mathcal{A}_n$ has less than 2^n states, there are two sets $S_1, S_2 \subseteq \Sigma_n^*$ such that the words w_{S_1} and w_{S_2} lead to the same state in $\mathcal{A}_n$. Formally, $\delta^*(q_0, w_{S_1}) = \delta^*(q_0, w_{S_2}) = q$, for some state q. Let $\sigma \in \Sigma_n^*$ be such that, without loss of generality, $\sigma \in S_1 \setminus S_2$. Consider the state $q' = \delta^*(q, \sigma \cdot \#)$. On the one hand, as $(w_{S_2} \cdot \sigma \cdot \#)^\omega \notin L_n$, we have that $q' \notin \alpha$. On the other hand, as $(w_{S_1} \cdot \sigma \cdot \#)^\omega \in L_n$, and $\mathcal{A}_n$ is deterministic, we have that $q' \in \alpha$, and we have reached a contradiction. ◀

In the context of monotonicity, the construction of $\mathcal{A}^{\frac{1}{\omega}}$ enables us to lift the positive result about DFW[1]s to DBW[1]s. For this, we prove that the construction in Theorem 26 preserves weak[1]ness:

▶ **Theorem 28.** *DBW[1]s are EM.*

Proof. Consider a DBW[1] $\mathcal{A} = \langle \Sigma, Q, q_0, \delta, \alpha \rangle$. In order to use the notations in Theorem 26, we view α as a parity condition $\alpha : Q \to \{1, 2, 3\}$. Let $\mathcal{A}^{\frac{1}{\omega}} = \langle \Sigma, \mathcal{F}, \rho, f_0, \alpha' \rangle$ be the DFW obtained by applying the construction in Theorem 26 on $\mathcal{A}$. We prove that $\mathcal{A}^{\frac{1}{\omega}}$ is a DFW[1]. Then, EMness of $L(\mathcal{A})$ follows from the fact that $Pr^\omega(L(\mathcal{A}), n) = Pr(L(\mathcal{A}^{\frac{1}{\omega}}, n)$ and Theorem 12.

Consider a reachable state $f \in \mathcal{F}$. Let $u \in \Sigma^*$ be such that $\rho^*(f_0, u) = f$. Thus, f captures u. Consider a letter $\sigma \in \Sigma$, and let $f' = \rho(f, \sigma)$. Thus, f' captures $u \cdot \sigma$. We prove that if $f \neq f'$, then $f \neq \rho^*(f, \sigma \cdot w)$, for all $w \in \Sigma^*$. Thus, the only cycles through the state f are self loops.

Let $f = \langle s, c \rangle$, $f' = \langle s', c' \rangle$, and assume that $f' \neq f$. First, if for all states $q \in Q$, we have that $s(q) = s'(q)$, then, as $c'(q) = \min\{c(q), \alpha(s'(q))\}$, it must be that for all states $q \in Q$, we also have that $c(q) = c'(q)$. Thus, $f' \neq f$ implies that there is a state $q \in Q$ such that $s(q) \neq s'(q)$. Since f captures u and f' captures $u \cdot \sigma$, the latter implies that $\delta^*(q_0, u) \neq \delta^*(q_0, u \cdot \sigma)$. Hence, as $\mathcal{A}$ is a DBW[1], we have that $\delta^*(q_0, u) \neq \delta^*(q_0, u \cdot \sigma \cdot w)$ for all $w \in \Sigma^*$. Thus, $f \neq \rho(f, \sigma \cdot w)$, for all $w \in \Sigma^*$, and we are done. ◀

We now show that Theorem 28 is tight.

▶ **Theorem 29.** *DBW[2]s and NBW[1]s may be NM.*

Proof. Consider the DBW[2] obtained by viewing the DFW from Figure 6 (left) as a Büchi automaton with $\alpha = \{s_2\}$. Consider also the NBW[1] obtained by viewing the NFW[1] appearing in Figure 4 (right) as a Büchi automaton. It is easy to see that both Büchi automata recognize the language $comp((ab)^\omega)$. As argued in the proof of Theorem 23, the language is NM. ◀

7 Discussion

We studied how the length of words influences their probability to belong to a regular language. We characterized formalisms that induce monotonic languages. The characterization is tight, in the sense that all the restrictions composing it are essential. Nevertheless, the characterization is not a necessary condition, in the sense that there are languages that do not satisfy it and are monotonic. The general problem of deciding the monotonicity characteristics of the language of a given automaton is the subject of future research. As discussed in Section 3.3, the question is very easy for unary automata. Moreover, our construction of $\mathcal{A}^{\frac{1}{\omega}}$ solves the challenges that have to do with periodicity and extends a solution for the setting of finite words to a solution for the setting of infinite words. We showed that the answer to the question depends also in the distribution of the letters in the alphabet. Since the problem of finding the spectrum of a stochastic matrix is difficult, our problem is expected to be difficult too. A related problem is to find the length in which eventually monotonic languages start to be monotonic. As we have seen in Example 2, this distance may be exponential in the size of the automaton also for deterministic weak automata of width 1.

References

1. I. Beer, S. Ben-David, C. Eisner, and Y. Rodeh. Efficient detection of vacuity in ACTL formulas. *Formal Methods in System Design*, 18(2):141–162, 2001.
2. J. Berstel. Sur la densité asymptotique de langages formels. In *Proc. 1st Int. Colloq. on Automata, Languages, and Programming*, pages 345–358, 1972.
3. J. Berstel, D. Perrin, and C. Reutenauer. *Codes and Automata*, volume 129 of *Encyclopedia of mathematics and its applications*. Cambridge University Press, 2010.
4. M. Bodirsky, T. Gärtner, T. von Oertzen, and J. Schwinghammer. Efficiently computing the density of regular languages. In *Proc. 6th Latin American Symposium on Theoretical Informatics*, volume 2976 of *Lecture Notes in Computer Science*, pages 262–270. Springer, 2004.
5. M. Chechik, M. Gheorghiu, and A. Gurfinkel. Finding state solutions to temporal queries. In *Proc. Integrated Formal Methods*, 2007. To appear.
6. J. Cohen, D. Perrin, and J-Eric Pin. On the expressive power of temporal logic. *Journal of Computer and System Sciences*, 46(3):271–294, 1993.
7. S. Ben David and O. Kupferman. A framework for ranking vacuity results. In *11th Int. Symp. on Automated Technology for Verification and Analysis*, volume 8172 of *Lecture Notes in Computer Science*, pages 148–162. Springer, 2013.
8. R. Fagin. Probabilities in finite models. *Journal of Symb. Logic*, 41(1):50–5, 1976.
9. G. De Giacomo and M. Y. Vardi. Linear temporal logic and linear dynamic logic on finite traces. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pages 854–860, 2013.
10. Y.V. Glebskii, D.I. Kogan, M.I. Liogonkii, and V.A. Talanov. Range and degree of realizability of formulas in the restricted predicate calculus. *Kibernetika*, 2:17–28, 1969.
11. Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.
12. O. Kupferman and M.Y. Vardi. Model checking of safety properties. *Formal Methods in System Design*, 19(3):291–314, 2001.
13. R. McNaughton and S. Papert. *Counter-Free Automata*. MIT Pres, 1971.
14. Y. Nakamura. The almost equivalence by asymptotic probabilities for regular languages and its computational complexities. In *Proc. 7th International Symposium on Games, Automata, Logics and Formal Verification*, volume 226 of *EPTCS*, pages 272–286, 2016.
15. A. Pnueli. The temporal semantics of concurrent programs. *Theoretical Computer Science*, 13:45–60, 1981.

16 Jeffrey S. Rosenthal. Convergence rates for markov chains. *SIAM Review*, 37(3):387–405, 1995.

17 Ben-David S, D. Fisman, and S. Ruah. Temporal antecedent failure: Refining vacuity. In *Proc. 18th Int. Conf. on Concurrency Theory*, volume 4703 of *Lecture Notes in Computer Science*, pages 492–506. Springer, 2007.

18 A. Salomaa and M. Soittola. *Automata Theoretic Aspects of Formal Power Series*. Springer-Verlag, 1978.

19 R. Sin'ya. An automata theoretic approach to the zero-one law for regular languages: Algorithmic and logical aspects. In *Proc. 6th International Symposium on Games, Automata, Logics and Formal Verification*, volume 193 of *EPTCS*, pages 172–185, 2015.

20 A.P. Sistla. Safety, liveness and fairness in temporal logic. *Formal Aspects of Computing*, 6:495–511, 1994.

21 P. Wolper. Temporal logic can be more expressive. *Information and Control*, 56(1–2):72–99, 1983.

A Useful Results from Linear Algebra

In this section we elaborate on the relevant results from linear algebra that are used in the proof of Theorem 11. The considerations are similar to these developed in [16]. For simplicity, we assume that all matrices we consider are squared (unless stated otherwise).

Recall that an *eigenvalue* of a matrix A is a scalar (denoted as λ_i) such that there exists a non-zero vector v for which $Av = \lambda_i v$.

The *characteristic polynomial* of A is the equation $det(A - \lambda I)$ where I is the identity matrix, det represent the determinant operation and λ is a scalar.

It is known that the eigenvalues of A are the roots of the characteristic polynomial (see [11], Chapter 1).

The *algebraic multiplicity* of an eigenvalue λ_i is its multiplicity as a root of the characteristic polynomial.

For two matrices A and B, we say that A is *similar* to matrix B if there exists an invertible matrix U such that $A = UBU^{-1}$. It can be shown that similar matrices have the same characteristic polynomials, as such they have the same eigenvalues.

A matrix is *diagonal* if all its non-zero elements are on the diagonal, and is *diagonalizable* if it is similar to some diagonal matrix.

A *Jordan block* is a matrix A with some $\lambda \in \mathbb{C}$ on the diagonal and 1's on the superdiagonal, thus A is of the following form

$$\begin{pmatrix} \lambda & 1 & 0 & 0 \\ 0 & \lambda & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & \lambda \end{pmatrix}.$$

The 1×1 matrix (λ) is also a Jordan block. A Jordan Block with $\lambda \in \mathbb{C}$ on its diagonal is sometimes called a *Jordan block of* λ.

A *Jordan matrix* is a matrix with Jordan blocks on its diagonal. In other words, let $\{B_1, \ldots, B_n\}$ be some Jordan blocks. Then, a Jordan matrix is of the following form

$$\begin{pmatrix} B_1 & 0 & 0 & 0 \\ 0 & B_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & B_n \end{pmatrix}.$$

Let A be some matrix. It is known that although not all matrices are diagonalizable, there always exists a Jordan matrix J that is similar to A (see [11], Chapter 3). Furthermore, the Jordan blocks of J are made of the eigenvalues of A and each block is of size that is not bigger than the algebraic multiplicity of its eigenvalue. Matrix J is called the *Jordan normal form* of A. Note that J is only unique up to the order of its Jordan blocks.

Jordan blocks give us a way to calculate the n'th power of a matrix directly. Let B be a Jordan block of size k. Then, for $n > k$, we have that

$$B^n = \begin{pmatrix} \lambda^n & \binom{n}{1}\lambda^{n-1} & \binom{n}{2}\lambda^{n-2} & \cdots & \cdots & \binom{n}{k-1}\lambda^{n-k+1} \\ 0 & \lambda^n & \binom{n}{1}\lambda^{n-1} & \cdots & \cdots & \binom{n}{k-2}\lambda^{n-k+2} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \lambda^n & \binom{n}{1}\lambda^{n-1} \\ 0 & \cdots & \cdots & \cdots & 0 & \lambda^n \end{pmatrix}. \tag{1}$$

▶ **Example 30.** Recall that in Example 9 we considered the DFW $\mathcal{A}$ from Example 8 and calculated the probabilities x_0^n and x_1^n of a run of length n to end on the states q_0 and q_1, respectively. We can now elaborate on the calculation, which is based on diagonalizing $P_{\mathcal{A}}$:

$$P_{\mathcal{A}} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} \frac{-1}{2} & 1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} \frac{-1}{2} & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \frac{-2}{3} & \frac{2}{3} \\ \frac{2}{3} & \frac{1}{3} \end{pmatrix}$$

Accordingly,

$$\begin{aligned} \begin{pmatrix} x_0^n & x_1^n \end{pmatrix} &= \begin{pmatrix} 1 & 0 \end{pmatrix} \cdot P_{\mathcal{A}}^n \\ &= \begin{pmatrix} 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} \frac{-1}{2} & 1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} \frac{-1}{2} & 0 \\ 0 & 1 \end{pmatrix}^n \cdot \begin{pmatrix} \frac{-2}{3} & \frac{2}{3} \\ \frac{2}{3} & \frac{1}{3} \end{pmatrix} \\ &= \begin{pmatrix} \frac{2+\left(\frac{-1}{2}\right)^n}{3} & \frac{1-\left(\frac{-1}{2}\right)^n}{3} \end{pmatrix}. \end{aligned} \tag{2}$$

⌟

Let $M \in \mathbb{R}^{d\times d}$ be a square matrix. Let $\lambda_1, \lambda_2, \dots, \lambda_l$ be the distinct eigenvalues of M in descending order by their absolute value. That is, $|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_l|$. If M is diagonalizable, then for every index (i, j) in M, we have that

$$(M^n)_{ij} = c_1(\lambda_1)^n + c_2(\lambda_2)^n + \cdots + c_l(\lambda_l)^n, \tag{3}$$

where $c_1, c_2, \dots, c_l \in \mathbb{C}$ are complex numbers that depend on (i, j). If M is not diagonalizable, then we instead rely on the Jordan normal form of M, which always exists. Using Property (1), where $n > d$, we replace each expression $c_i(\lambda_i)^n$ in (3) by

$$(c_i^1(n)^{k_i-1} + c_i^2(n)^{k_i-2} + \cdots + c_i^{k_i})(\lambda_i)^n,$$

where k_i is the size of the largest Jordan block of λ_i, and $c_i^1, c_i^2, \dots, c_i^{k_i} \in \mathbb{C}$.

By [11] (Theorem 3.1.11), if all the eigenvalues of M are real, then $c_1^1, \dots, c_1^{k_1}, \dots, c_l^1, \dots, c_l^{k_l}$ are real too.

Let $P \in \mathbb{R}^{d\times d}$ be some stochastic matrix. It is shown in [16] that for all $1 \leq i \leq l$, we have that $|\lambda_i| \leq 1$, and that $\lambda_1 = 1$ is an eigenvalue of P. Furthermore, the largest Jordan block of $\lambda_1 = 1$ is of size 1.

We can now prove Lemma 10, which is key to the proof of Theorem 11.

▶ **Lemma 10.** *Consider the stochastic matrix $P \in \mathbb{R}^{d\times d}$ and the set $S = \{(i_1, j_1), \cdots, (i_t, j_t)\}$ of t indices in P. If all the eigenvalues of P are real and non-negative, then the sequence $\{\sum_{(i',j')\in S} P^n_{i',j'}\}_{n=1}^{\infty}$ is EM.*

Proof. Let $\lambda_1, \lambda_2, \dots, \lambda_l$ be the distinct eigenvalues of P in descending order by their absolute value. Let $\{a_n\} = \{\sum_{(i',j')\in S} P^t_{i',j'}\}_{t=1}^{\infty}$. Since P is a stochastic matrix, then for every $1 \le i \le l$, we have that $|\lambda_i| \le 1$. By our assumption, all the eigenvalues are real and non-negative, and so, for every $1 \le i \le l$, we have that $0 \le \lambda_i \le 1$. Since the largest Jordan block of $\lambda_1 = 1$ is of size 1, then for every $n > d$, we get that

$$a_n = (1)^n c_1^1 + (\lambda_2)^n((n)^{k_2-1}c_2^1 + (n)^{k_2-2}c_2^2 + \cdots + c_2^{k_2}) + \cdots + (\lambda_l)^n((n)^{k_l-1}c_l^1 + \cdots + c_l^{k_l}),$$

for some $c_1^1, c_2^1, \dots, c_2^{k_2}, c_3^1, \dots, c_l^1, \dots, c_l^{k_l} \in \mathbb{R}$.

Now, observe that when $n > d$, the difference between two successive iterations of the sequence is

$$a_n - a_{n+1} = (\lambda_2)^n((n)^{k_2-1}(c_2^1 - c_2^1\lambda_2) + \cdots + c_2^{k_2} - c_2^{k_2}\lambda_2) + (\lambda_3)^n(\cdots) + \cdots + (\lambda_l)^n(\cdots) \quad (4)$$

If Equation (4) is equal to zero, we have that $a_n = a_{n+1}$ and so $\{a_n\}$ is EM. Else, it is not hard to prove that since $1 > \lambda_2 > \lambda_3 > \cdots > \lambda_l \ge 0$, then there exists some non-zero element $(\lambda_i)^n(n)^{k_i-j}(c_i^j - c_i^j\lambda_i)$ that is dominant in Equation (4). That is, there exist $m_0 \ge d$ such that for all $m \ge m_0$, we have that

$$|(\lambda_i)^n(n)^{k_i-j}(c_i^j - c_i^j\lambda_i)| \ge |a_n - a_{n+1} - (\lambda_i)^n(n)^{k_i-j}(c_i^j - c_i^j\lambda_i)|. \quad (5)$$

Note that since $\lambda_i > 0$, we have that $(\lambda_i)^n(n)^{k_i-j} > 0$. Since c_i^j can be any real number, we need to distinguish between cases. If $c_i^j - c_i^j\lambda_i > 0$, then $a_n - a_{n+1} > 0$, implying that $\{a_n\}_{m_0}^{\infty}$ is MD. If $c_i^j - c_i^j\lambda_i < 0$, we have that $a_n < a_{n+1}$, in which case $\{a_n\}_{m_0}^{\infty}$ is MI. In both cases, $\{a_n\}$ is EM. ◀

Note that Example 8 does not contradict Theorem 11, as $\lambda = -\frac{1}{2}$ is an eigenvalue of P.