MEASURES INDUCED BY AUTOMATA AND THEIR ACTIONS

A Dissertation

by

ROMAN KOGAN

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Rostislav Grigorchuk
Committee Members,	Sergiy Butenko
	Volodymyr Nekrashevich
	Yaroslav Vorobets
Head of Department,	Emil Straube

December 2017

Major Subject: Mathematics

ABSTRACT

In this thesis we explore the theme of automata, measures on spaces of sequences $X^{\mathbf{N}}$ in a finite alphabet X, and their connections. The notion of a finite-state measure (a measure given by a finite automaton, or equivalently, having a finite number of sections) is introduced, and applied to the problem of studying the images of Markov measures under the action of tree automorphisms given by automata. Another approach, based on prior work by Kravchenko, is also applied to this problem to compute the Radon-Nikodym derivative in the case when the automaton has polynomial growth, and to compute frequencies by using a lift to $(S \times X)^{\mathbf{N}}$.

The question of when the image of a finite-state measure under the action of a non-invertible automaton is answered. We also explore when a finite-state measure is Gibbs.

For the second part of the thesis, we introduce the notion of the automatic logarithm, and a measure associated with it. We compute this measure for certain interesting examples, in which it turns out to be finite-state.

DEDICATION

To Monique Stewart,

the secretary of the Department of Mathematics of Texas A& M University.

in appreciation of all the hard work done for our sake.

ACKNOWLEDGMENTS

It's been a long way, and many people have supported me to make this thesis happen. I would like to thank many people, and I apologize to those whom I won't mention directly or otherwise - you can rest assured that the only reason for doing so is this thesis being typed days before submission, as you would likely expect of me. So, I'll try to thank, in semi-chronological order:

- my parents, and, in particular, my mother, who has introduced me to mathematics at an early age and sparked the ongoing interest in it;
- all who taught me math from elementary school through graduate school;
- Dmytro Karabash & Val Zakharevich for the influence
- Jim Kiernan, Murrow HS Math Team coach and independent learning coordinator
- Michael Bender, whose was the first to let me make undergraduates suffer through my problemsets, and all the professors for whom I ever was a TA or grader
- Dmitry Ryabogin, who organized KSU REU which made me swtich to math from CS (and through which I met Olena);
- All who advised me on (other) research projects throughout the years (David Yetter, Collin Bleak, Franesco Matucci, Sergiy Tabachnikov, Alexander Kirillov Jr., Olga Plamenvskaya, Maurice Rojas)
- Collin Bleak, who introduced me to the Grigorchuk Group and geometric group theory;

- YGGT conference, which finally convinced me to switch to GGT, and Krzysztof Swiecicki, who suggested I go there

I am grateful for the support of all the friends I made in the mathematics department here: Ngoc Do, Rishika Rupam, Dilber Kocak, Shahina Rahman, Fatma Terzioglu, Taylor Bruscewicz, Davis Sykes, Ben Francisco, Amanda Hosington, Krzysztof, and others.

Friends I made at math conferences: Kaidi Ye, Martina Aaltonen (who helped me dare start anew), Amanda Taylor, Rami Luisto, Vera Tonic, Thibault Godin, Waltraud Lederle, Tara Brough, Mark Hagen, Ana Claudia Lopez, Lorna Willington, Ricky Steinberg, Maria Cabello, and many others. Your bright personalities are a big part of what makes mathematics enjoyable. As Collin once told me, math - like food - is best shared.

Additionally, I want to thank all my friends who helped me make it through grad school, especially those in Bryan/College Station. You know who y'all are!

I wouldn't be here without the endless encouragement and support of my wife, Olena Ostapyuk, in mathematical and life pursuits over the last decade.

And finally, I am greatly indebted to Prof. Rostislav Grigorchuk and Prof. Yarsolav Vorobets whose guidance made this thesis a reality.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Professor Rostislav Grigorchuk (advisor) of the Department of Mathematics, Professor Sergiy Butenko of the Department of Industrial Engineering, and Professors Volodymyr Nekrashevich and Yaroslav Vorobets of the Department of Mathematics.

The chapters on automatic logarithm and images of Markov measure reference ongoing work with Grigrorchuk and Vorobets.

All the work conducted for the dissertation, unless otherwise indicated, was completed by the student independently.

Funding Sources

Graduate study was supported by a teaching assistantship from Texas A&M University.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

1. INTRODUCTION

This disseration contains several related projects on the theme of automata and measures on the spaces of sequences.

Finite automata are objects fundamental to computer science, which have found surprising applications in group theory in pioneering work of Rostislav Grigorchuk, providing a rich source of counter-examples (particularly, providing an example of a group with subexponential activity, answering the question of Milnor).

It turns out that a wide and interesting class of measures, called finite-state measures, can also be defined by finite automata. This class is preserved by the action of invertible finite automata, and contains measures such as Bernoulli, Markov, and k-step Markov measures. The problem of studying images of measures under the action of automata, initiated in [1] and [2] and later continued in [3], can thus be reduced to the study of properties of finite-state measures in the case when the automata are invertible. We examine when such measures are equivalent to k-step Markov measures, when they are Gibbs, and establish conditions for the image of a finite-state measure to be finite-state under the action of non-invertible automata.

Additionally, we extend the approach of [3] to study the images of Markov measures. This approach involves lifting, and is not equivalent to studying the properties of the finite-state measure corresponding to the image.

Another application of finite-automata in computer science and group theory is to construct the so-called graphs of action (also known as Schreier graphs, being a particular case of thereof). The problem of studying the distribution of lengths of chords in the graph of action of two initial automata on the levels of binary tree gave rise to the *automatic logarithm*, a map defined by an automaton that outputs these lengths. The distribution of the lengths of chords is then readily seen as the image of the uniform Bernoulli measure by the action of the automatic logarithm. The most interesting cases arise when the automatic logarithm is not invertible (the distribution is uniform otherwise), and the resulting measure is not necessarily finite state. Examples for important automata and either case are provided.

Finally, we conclude with remarks and open questions related to finite-state measures, their images, and Schreir graphs of automata.

2. PRELIMINARIES

2.1 Finite automata and related concepts

2.1.1 Endomorphisms of rooted trees

The finite **alphabet** X of size d is a finite set of cardinality d. We will use the following alphabets: the set $\{0, 1, \ldots, d-1\}$, and finite subsets $X \subset [0, 1]$.

For w - a word in X, $|w|$ denotes its length, and w_i denotes the i'th character for $0 \leq i \leq |w| - 1$. If v is another word (or a character), wv is the concatenation of the two, so $w = w_0 w_1 \ldots w_{|w|-1}$.

X^* denotes all finite words in X:

$$X^* := \{a_0 \ldots a_{n-1} \ : \ a_i \in X, n \in \mathbf{N} \cup \{0\}\}$$

A d**-regular, rooted tree** $\mathcal{T}$ is a rooted graph with the vertex set $V = X^*$, root at the empty word, and the edge set

$$F = \{(w, wa) \ : \ w \in X^*, a \in X\}.$$

An **endomorphism** of the tree $\mathcal{T}$ is a map from $\mathcal{T}$ to itself that is also a graph homomorphism (preserves the adjacency relation). An **automorphism** is an invertible endomorphism.

The n'th level X^n of a tree is the set of words of length n. Observe that endomorphisms, by definition, preserve levels.

The **boundary** of the tree $\mathcal{T}$ is the set $X^{\mathbf{N}}$ of infinite sequences in X:

$$\partial T := \{a_0 a_1 a_2 \ldots \ : \ a_i \in X, i \in \mathbf{N}\}$$

∂T is supplied with the Tychonoff product topology that makes it homeomorphic to a Cantor set.

Let σ_r denote the operation that deletes the last character of a word: for $w \in X^*$ and $a \in X$,

$$\sigma_r(wa) := w.$$

Remark 2.1.1. *∂T can be obtained as the inverse limit of the direct system of levels $\{X^n : n \in \mathbf{N}\}$ with the projections $\psi_{m,n} : X^n \to X^m$ given by $\psi_{m,n} := \sigma_r^{n-m}$ (i.e., discarding the last $n - m$ characters).*

2.1.2 Mealy and Moore machines

Definition 2.1.2: A **Mealy machine**, or a **finite initial automaton with output**, is a hextuple $\mathcal{A}_q = (S, q, X, Y, \pi, \lambda)$, with

- X - a finite (input) alphabet;
- Y - a finite (output) alphabet;
- S - a finite set of *states*;
- $q \in S$ - the *initial state*;
- $\pi : S \times X \to S$ - the *transition map*
- $\lambda : S \times X \to Y$ - the *output map*

After specifying the initial state q of the automaton $\mathcal{A}_q$, we often omit it in further mentions, and simply write $\mathcal{A}$ for $\mathcal{A}_q$.

We write π_s, λ_s for restrictions of these functions to the state s, defining $\pi_s(x) := \pi(s, x)$ and $\lambda_s(x) := \lambda(s, x)$.

The functions π and λ also act on words in the alphabet X via these recursive definitions (for $x \in X$, $w \in X^*$):

$$\pi(s, xw) := \pi(\pi(s, x), w);$$
$$\lambda(s, xw) := \lambda(s, x)\lambda(\pi(s, x), w).$$

In the same way, π_s and λ_s, for $s \in S$, act on words $w \in X^*$. Additionally, we may write $\pi(w)$ for $\pi_q(w)$ (and similarly, $\lambda(w)$ for $\lambda_q(w)$) when q is the initial state.

We write

$$s_0 \xrightarrow[y_1]{x_1} s_1 \xrightarrow[y_2]{x_2} \ldots \xrightarrow[y_n]{x_n} s_n$$

when the automaton takes an input string $x_1x_2\ldots x_n$, outputs $y_1y_2\ldots y_n$ while going through a sequence of states $s_0s_2\ldots s_n$. That is, $\pi(s_{i-1}, x_i = s_i)$ and $\lambda(s_{i-1}, x_i = y_i)$ for $i \in 1..n$. We call the above a **path** in the automaton (as it is indeed a path in the directed graph where the edges are given by the set of states S, the transition function π providing the incidence matrix, and λ giving edge labels).

An automaton A is **invertible** if $X = Y$ and λ is injective. To an invertible automaton A with states $s_1, \ldots, s_n$, we associate an automaton A^{-1} with states $s_1^{-1}, \ldots, s_n^{-1}$ defined as follows:

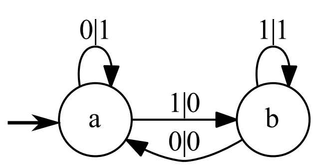
$$\lambda(s_i^{-1}, \lambda(s_i, x)) := x;$$
$$\pi(s_i^{-1}, \lambda(s_i, x)) := \pi(s_i, x).$$

This is well-defined, and for $w \in X^{\mathbf{N}}$, $s_i^{-1} \circ s_i(w) = s_i \circ s_i^{-1}(w) = w$. Since the actions preserve word length, they are isomorphisms of the regular tree where the children of every node are indexed by X. We can now talk about the **group of tree automorphisms** generated by the states of A.

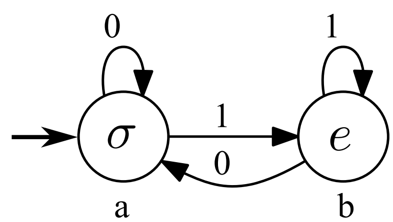
We call an automaton A **strongly-connected** if for every pair of states $s, t \in S$ there exists a path that starts in s and ends in t. A tree automorphism is **strongly-connected** whenever its automaton of restrictions is.

We define a stronger notion of connectedness (L**-strongly-connectedness**) in a later section, which we use as a sufficient condition for ergodicity of certain measures.

The **diagram** of an automaton $\mathcal{A}_q$ is a labeled graph with the vertex set S, edge set $E = \{(s, \pi(s,x)) \ : \ s \in S, x \in X\}$, with label $x : \lambda(s,x)$ on the edge $(s, \pi(s,x))$. The initial state q is marked with an arrow. An example of such diagram for the Lamplighter automaton **L** is shown in Figure 2.1a.



(a) Diagram with output marked on edges



(b) Diagram with element of the symmetric group on vertices

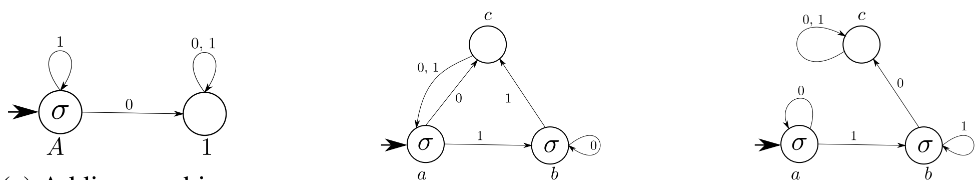
Figure 2.1: Two ways to draw the Lamplighter automaton

An automaton $\mathcal{A}$ is **invertible** if λ_s is invertible for all $s \in S$ (that is, if $\lambda_s \in S(X)$, where $S(X)$ is the symmetric group on X). The endomorphism g given by an invertible automaton A is invertible, and the automaton for g^{-1} (which we denote as A^{-1}) can be constructed from the diagram of A by flipping the input and output on the edges.

In the case when an automaton is invertible, we can draw the diagram of the automaton without specifying its output on the arrows. Instead, the state s is marked by $\lambda_s \in S(X)$.

If λ_s is the trivial permutation, we call the state s **passive**, and call it **active** otherwise.

When $X = \{0, 1\}$, we write σ for the nontrivial permutation of X (i.e. $\sigma(0) = 1, \sigma(1) = 0$). In the diagrams of automata over $X = \{0, 1\}$ we then mark active states with σ, leave the label of passive states blank. Figure 2.1b shows how to draw the Lamplighter automaton of Figure 2.1a in this way. A few more examples of such diagrams are in Figure 2.2, and further throughout this paper.



(a) Adding machine (b) automaton F (c) automaton Z

Figure 2.2: Diagrams of invertible automata

Unless otherwise specified, we assume $X = Y$, and write an automaton $\mathcal{A}_q = (S, q, X, \pi, \lambda)$.

An automaton state q acts on X^*, the d-ary tree $\mathcal{T}$, and its boundary ∂T by the action of A_q. We shall use A and q interchangeably for this action when the context is clear.

With an invertible tree endomophism g we can associate a **portrait** diagram that uniquely determines g. Note that for a finite word w, $g|_w$ acts on X by a permutation when g is invertible. The portrait consists of the infinite tree $\mathcal{T}$ with markings on the nodes: node corresponding to word w is marked with the permutation of X induced by $g|_w$. When $|X| = 2$, we only mark the nodes with nontrivial permutation, and leave others unmarked.

Example 2.1.3. *The portrait of the adding machine of Figure 2.2a is shown in Figure 2.3.*

Remark 2.1.4. *Every tree automorphism has a portrait, but not all tree automorphisms are given by finite automata.*

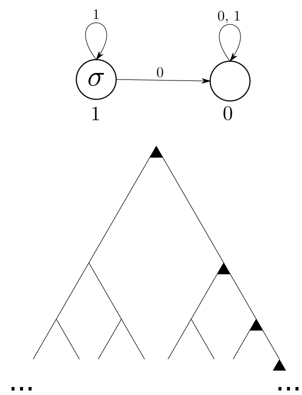


Figure 2.3: The adding machine and its portrait

Definition 2.1.5: A Mealy automaton is said to be a **Moore machine** when the output does not depend on the input. That is, for all $s \in S$, λ_s is constant: for all $x, y \in X$, $\lambda(s, x) = \lambda(s, y)$. In this case, we simply write $\lambda(s)$ for the value λ_s takes.

Remark 2.1.6. *In this definition, the output only depends on the current state s. Some authors use the definition of a Moore machine with a shift, where the output is determined by the ending state $\pi(s, x)$, and so does depend on the input.*

Mealy automata A and B are said to be **equivalent** if $A(w) = B(w)$ for all $w \in X^*$.

Definition 2.1.7: An initial Mealy automaton A is said to be **minimal** if it has the smallest number of states out of all the automata in its equivalence class. This is a classical notion, as is the minimzation algorithm that produces the minimal automaton in a given class; see [4] for a discussion of this algorithm (refer to [5] for a discussion of this equivalence and an algorithm for constructing the minimal automaton in the more general case of asynchronous Mealy machines, which we do not consider here). We also present an implementation of an algorithm that minimizes a Mealy machine in the appendix.

Given automata A and B such that the output alphabet of A coincides with the input alphabet of B, one can construct the **product automaton**, denoted $A \cdot B$, which computes the composition $A \circ B$. We again refer to [5] for the construction of the product automaton.

2.1.3 Sections of tree endomorphisms

Definition 2.1.8: Let g be an endomorphism of a d-regular rooted tree $\mathcal{T}$, and w - a finite word. A **section** of g by w, denoted $g|_w$, is an endomorphism h of $\mathcal{T}$ such that for any word or sequence v, $g(wv) = g(w)h(v)$.

Remark 2.1.9. *by definition of the automaton, and using the notation above, a finite automaton A has a finite set of sections, which is the subset of its states in the connected component of the starting state in the diagram of the automaton $\mathcal{A}$.*

To a tree endomorphism g we can associate an automaton $A = (S, g, X, \pi, \lambda)$ with the initial state labeled by g, such that the action of A is identical to the action of g. We take $S = \{g|_w \ : \ w \in X^{\mathbf{N}} \sqcup g\}$, and define $\pi(h, x) := h|_x$; $\lambda(h, x) = h(x)$. The automaton of restriction, in general, needs not be finite. When it is finite, the tree automorphism g is said to be **finite-state**.

Remark 2.1.10. *An automorphism g of the tree $\mathcal{T}$ is finite-state if and only if its portrait contains a finite number of distinct (up to isomorphism of marked trees) subtrees. The subtrees in the portrait diagram define sections of g.*

We now prove several basic propositions related to sections of automorphisms which we use in subsequent chapters.

Proposition 2.1.11. *When an endomorphism g is invertible, all of its sections are invertible, and for $w \in X^*$, $(g|_w)^{-1} = g^{-1}|_{g(w)}$.*

Proof. Let $w \in X^*$ and $v \in X^{\mathbf{N}}$. Then by definitions,

$$\begin{aligned} wv &= g^{-1}(g(wv)) \\ &= g^{-1}(g(w)g|_w(v)) \\ &= g^{-1}(g(w))g^{-1}|_{g(w)}(g|_w(v)) \\ &= wg^{-1}|_{g(w)}(g|_w(v)). \end{aligned}$$

Therefore, $g^{-1}|_{g(w)}(g|_w(v)) = v$, and the proposition holds. □

Proposition 2.1.12. *Let A be a tree endomorphism, and w, v be finite words in X. Then $A|_{wv} = (A|_w)|_v$.*

Proof. For u - any word, $A(wvu) = A(w)A|_w(vu) = A(w)A|_w(v)(A|_w)|_v(u) = A(wv)(A|_w)|_v(u)$; so the proposition holds by definition. □

Proposition 2.1.13. *Let A, B be tree endomorphisms, and w - finite word in X. Then $(AB)|_w = A|_{B(w)}B|_w$.*

Proof. Let v be a finite word. By the definition of section,

$$\begin{aligned} AB(wv) &= A(B(w)B|_w(v)) \\ &= AB(w)A|_{B(w)}B_w(v), \end{aligned}$$

so $(AB)|_w = A|_{B(w)}B_w$. □

Corollary 2.1.14. *Let A, B be tree endomorphisms, and w, v - finite words in X. Then $(AB)|_{wv} = A|_{B(wv)}B|_{wv} = A|_{B(w)B|_w(v))}(B|_w)|_v$.*

Proposition 2.1.15. $A^n|_w = A|_{A^{n-1}(w)}A|_{A^{n-2}(w)} \dots A|_{A(w)}A|_w$.

Proof. The result holds trivially when $n = 1$. By Prop. 2.1.13,

$$\begin{aligned} A^n|_w &= (A \circ A^{n-1})|_w \\ &= A|_{A^{n-1}(w)}(A^{n-1}|_w). \end{aligned}$$

The result follows by induction. □

Proposition 2.1.16. *Assume A acts transitively on levels, $|w| = n$, and $a \in X$. Then $A^{2^n}|_w(a) \neq a$.*

Proof. If $A^{2^n}|_w(a) = a$, then wa, a word of length $n+1$, is a fixed point of A^{2^n}, contrary to the assumption that the length of the orbit of A on words of length $n+1$ is 2^{n+1}.

2.1.4 Bounded-activity automata

Definition 2.1.17: An automaton A is said to have **bounded activity** if the number of nontrivial sections on every level is bounded by a global constant c:

$$\exists c : \forall n \in \mathbf{N} : \ |\{A|_w \ : \ A|_w \neq \mathbf{1}, w \in X^n\}| < c.$$

In such cases, we say that the automaton A is **bounded**.

Example 2.1.18. *The adding machine in Figure 2.2a is a bounded automaton.* △

Proof. The two states of the automaton are the active state, A (nontrivial section), and the trivial state, $\mathbf{1}$ (trivial section). From the diagram of the automaton:

$$A|_0 = \mathbf{1}$$

$$A|_1 = A$$

$$\mathbf{1}|_0 = \mathbf{1}$$

$$\mathbf{1}|_1 = \mathbf{1}$$

By inductively applying Prop. 2.1.12, $A|_w \neq \mathbf{1}$ if and only if $w = 11\ldots1$. Thus the number of nontrivial sections on every level is 1, and the adding machine is a bounded automaton. $\triangle$

Remark 2.1.19. *The adding machine can also be defined by its portrait in Figure 2.3, in which case it is bounded by definition.*

Proposition 2.1.20. *If A is a tree endomorphism given by a finite Mealy automaton $\mathcal{A}$ which is bounded and acts transitively on levels, then the set*

$$T_A := \{A^n|_w \ : \ n \leq 2^{|w|}, w \in X^*\}$$

is finite. That it, the set of sections of powers of a bounded automaton is finite.

Proof. First, from Prop. 2.1.15,

$$T_A := \{A|_{A^{n-1}(w)}A|_{A^{n-2}(w)}\ldots A|_{A(w)}A|_w \ : \ n \leq 2^{|w|}, w \in X^*\}.$$

For a given w, consider the finite sequence of words $w, A(w), A^2(w), \ldots, A^{n-1}(w)$ with $n \leq 2^{|w|}$. By transitivity of action of A, all elements in it are distinct, and thus this sequence is a subset of vertices on level $|w|$.

Since A is bounded, let c be the constant such that at most c sections on every level are nontrivial. Thus in any product

$$A|_{A^{n-1}(w)}A|_{A^{n-2}(w)}\ldots A|_{A(w)}A|_w$$

with $n \leq 2^{|w|}$, at most c elements are nontrivial. Since A is finite by assumption, the nontrivial sections are enumerated by the finite set of states S_A of $\mathcal{A}$. Therefore,

$$|T_A| \leq |S_B|^c. \square$$

2.1.5 Activity of tree automorphisms

A tree automorphism g is said to have **polynomial activity** (resp. subexponential, exponential activity) if the number of words w of length n such that $g|_{wX^{\mathbb{N}}}$ is nontrivial is at most polynomial (resp. subexponential, exponential) in n. That is, the number of nontrivial sections grows polynomial in level.

If g has polynomial activity, then $g|_w$ is trivial for most w. For such w, $\hat{g} = \pi(g, w)$ acts trivially; we call such states **trivial**, and all others - **nontrivial**. Note that in the diagram of that automaton, all paths going from a trivial state can only go to trivial states; so if the automaton is reduced, it may only have at most one trivial state I, with $\pi(I, x) = I$ and $\lambda(I, x) = x$ for all $x \in X$.

From the above, it follows that if g is a tree automorphism generated by a reduced finite automaton A (with initial state g), then A has a unique trivial state.

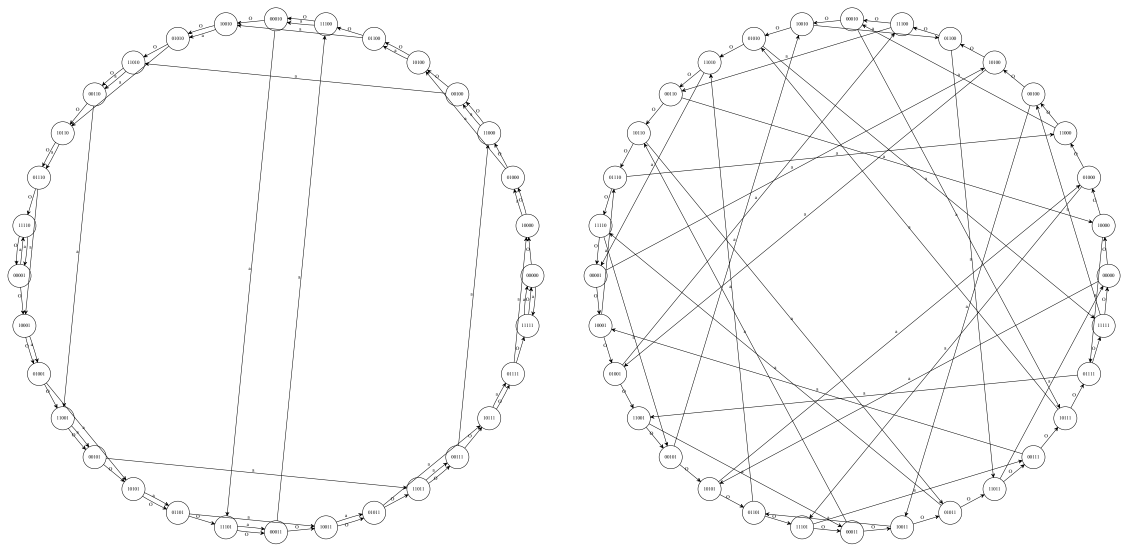
It is easy to show that strongly-connected automata, in general, do not have polynomial activity (e.g. strongly-connected automata with two or more nontrivial cycles). Thus the strongly-connected automata are a counterpart to the polynomial-activity ones.

2.1.6 Graphs of action

Definition 2.1.21: The **graph of the action** of an initial automaton $\mathcal{A}_q = (S, q, X, \pi, \lambda)$ on a set $S \subset \mathcal{T}$ is the directed graph with vertex set S and edges $w \to \lambda_q(w)$ for $w \in S$. The graph of action of automata $\mathcal{A}_{1_{q_1}}, \mathcal{A}_{2_{q_2}}, \ldots, \mathcal{A}_{k_{q_k}}$ on S is similarly defined as a directed graph with vertex set S and edges $w \to \lambda_{i_{q_i}}(w)$, $1 \leq i \leq k$ and $w \in S$.

In this paper, we consider **graphs of action on level** n of two automata, $\mathcal{O}$ and $\mathcal{A}$, with

$\mathcal{O}$ being the adding machine (Figure 2.2a). Figure 2.4 shows examples of such graphs for $\mathcal{A}$ being automaton Z (Figure 2.2c) and $\mathcal{A}$ being automaton F (Figure 2.2b).



(a) The graph of action of adding machine and automaton Z

(b) The graph of action of adding machine and automaton F

Figure 2.4: Examples of Schreier graphs

2.2 Measure-theoretic definitions

2.2.1 Ergodic theory

We now give a few definitions relevant to probability theory (ergodic theory).

A **cylinder set** $wX^{\mathbf{N}}$ is a clopen subset of $X^{\mathbf{N}}$ given by

$$wX^{\mathbf{N}} := \{wv \ : \ w \in X^*, v \in X^{\mathbf{N}}\}.$$

A **probability vector** p is a a vector $p : X \to [0, 1]$ with $\sigma_{i \in X} p(i) = 1$. A **stochastic matrix** on X is a matrix $M : X \times X \to [0, 1]$ whose rows are probability vectors.

Definition 2.2.1: The **Bernoulli measure** on $X^{\mathbf{N}}$ is a probability measure defined by a probability vector p, given on the cylinders $wX^{\mathbf{N}}$ by

$$\mu(wX^{\mathbf{N}}) := \prod_{i=0}^{|w|-1} p(w_i),$$

and extended by additivity properties op probability measures. The **uniform Bernoulli measure** is given by $p = \left(\frac{1}{|X|}, \dots, \frac{1}{|X|}\right)$ (so, in our case, by $p = \left(\frac{1}{2}, \frac{1}{2}\right)$.

Informally, this measures probability of a sequence of independent events (e.g. coin flips).

Definition 2.2.2: The **Markov measure** given by a probability vector l and a stochastic matrix L is given on the cylinder sets $wX^{\mathbf{N}}$ by

$$\mu(wX^{\mathbf{N}}) := l(w_0) \prod_{i=1}^{|w|-1} L_{w_{i-1},w_i}.$$

Informally, this measures the probability of events where the probability of an outcome may depend on what the preceding outcome was.

Let $X = \{a_1, \dots, a_k\}$ be a finite alphabet, and L-a k-by-k **irreducible stochastic matrix** (with entries indexed by X) with a **stationary probability vector** l. That is, $\sum_b L_{ab} = 1$ for all $a \in X$, $lL = l$, and the directed graph whose incidence matrix is given by nonzero entries of L is path-connected. Let μ be the **invariant Markov measure** on $X^{\mathbf{N}}$ induced by L; μ is given on the cylinders by

$$\mu(x_1 x_2 \dots x_n X^{\mathbf{N}}) = l_{x_1} L_{x_1,x_2} L_{x_2,x_3} \dots L_{x_{n-1}x_n}.$$

In this text, we consider the following generalization of Markov measures, and compare it to finite-state measures (to be defined later):

Definition 2.2.3: a **Bowen-Gibbs measure**, or simply a **Gibbs** measure ν is a measure such that for some $C, P \in \mathbf{R}$ ($C > 0$) and all $w \in X^n n$,

$$\frac{1}{C} < \frac{\nu(w_0 \dots w_{n-1} X^{\mathbf{N}})}{exp(-nP + \sum_{i=0}^{n-1} f(\sigma^i w))} < C.$$

The function f is called the **potential** of the measure ν. For further details, refer to [6] for an introduction to the subject.

Hereafter, we let $\mathcal{B}$ be the sigma-algebra generated by cylinder sets wX^∞, for w - finite word.

The **shift operator** σ acts on $X^{\mathbf{N}}$ by "eating" the first character: $\sigma(aw) = w$, for $a \in X, w \in X^{\mathbf{N}}$.

A probability measure ν on $X^{\mathbf{N}}$ is **invariant** (with respect to the shift σ) if $\nu(\sigma^{-1}(E)) = \nu(E)$ for all $E \in \mathcal{B}$. We call an probability measure ν **ergodic** if for all **invariant sets** $E \in \mathcal{B}$ (sets such that $\sigma^{-1}(E) = E$), either $\nu(E) = 1$ or $\nu(E) = 0$ holds.

Equivalently, we may say that the shift σ is an invariant and ergodic transformation on $(X, \mathcal{B}, \nu)$, or that $(X^{\mathbf{N}}, \mathcal{B}, \nu)$ is a **probability-preserving transformation** (ppt for short).

Example 2.2.4. *the invariant Markov measure μ defined above is invariant and ergodic [7]. It is invariant, but not ergodic if we let L be not irreducible. Bernoulli measures given by a positive vectors are specific cases of invariant Markov measures, and so are invariant and ergodic.* $\triangle$

2.2.2 Sections of a measure

Definition 2.2.5: the **null measure** ν_0, also called a **trivial measure**, is a measure defined by $\nu_0(E) = 0$ for all cylinder sets E.

Definition 2.2.6: If μ is a probability measure on $X^{\mathbf{N}}$, the **section** of μ by a word

$w \in X^*$ is $\mu|_w$ defined by

$$\mu|_w(vX^{\mathbf{N}}) := \begin{cases} \dfrac{\mu(wvX^{\mathbf{N}})}{\mu(wX^{\mathbf{N}})}, \text{ when } \mu(wX^N) \neq 0; \\ 0, \text{ otherwise} \end{cases}$$

for all $v \in X^*$. When μ is a probability measure, μ_w can be seen as the conditional probability given w.

We say a word w is **admissible** (with respect to μ) if $\mu(wX^{\mathbf{N}}) \neq 0$. Unless specified, we assume that w is admissible when the section $\mu|_w$ is taken. We say a word w is **forbidden** if it is not contained in any admissible word.

The following propositions outline how to compute sections of measures.

Proposition 2.2.7. *For $v, w \in X^*$, $(\mu|_w)|_v = \mu|_{wv}$.*

Proof. If $\mu(wX^{\mathbf{N}}) = 0$, then $\mu(wvX^{\mathbf{N}}) = 0$, and the proposition holds. Otherwise, assume $\mu(wX^{\mathbf{N}}) \neq 0$ and $\mu(wvX^{\mathbf{N}}) \neq 0$.

Let $u \in X^*$. By definition,

$$\begin{aligned} (\mu|_w)|_v(uX^{\mathbf{N}}) &= \frac{\mu|_w(vuX^{\mathbf{N}})}{\mu|_w(vX^{\mathbf{N}})} \\ &= \frac{\mu(wvuX^{\mathbf{N}})}{\mu(wX^{\mathbf{N}})\mu|_w(vX^{\mathbf{N}})} \\ &= \frac{\mu(wvuX^{\mathbf{N}})}{\mu(wvX^{\mathbf{N}})} \\ &= \mu|_{wv}(uX^{\mathbf{N}}). \ \square \end{aligned}$$

Corollary 2.2.8. *Let $\mu = \sum_{i=1}^{k} a_i\mu_i$, where $a_i \in \mathbf{R}$ and μ_i are probability measures, and* *let w be admissible. Then*

$$\mu|_w = \frac{1}{\mu\left(wX^{\mathbf{N}}\right)} \sum_{i=1}^{k} a_i \mu_i \left(wX^{\mathbf{N}}\right) \mu_i|_w.$$

Proof. For $v \in X^*$,

$$\mu|_w \left(vX^{\mathbf{N}}\right) = \frac{1}{\mu\left(wX^{\mathbf{N}}\right)} \sum_{i=1}^{k} a_i \mu_i \left(wvX^{\mathbf{N}}\right) = \frac{1}{\mu\left(wX^{\mathbf{N}}\right)} \sum_{i=1}^{k} a_i \mu_i \left(wX^{\mathbf{N}}\right) \mu_i|_w. \quad \square$$

3. FINITE-STATE MEASURES

In this chapter we introduce the notion of a finite-state measure, explore its properties, and study images of Markov and finite-state measures under the action of tree endomorphisms.

3.1 Definition and basic propositions

Definition 3.1.1: a measure μ is **finite-state** if the set of sections of μ is finite.

Example 3.1.2. *Bernoulli and Markov measures (definitions 2.2.1 and 2.2.1, resp.) are finite-state:*

- *a Bernoulli measure μ only has one (nontrivial) section, since when w is admissible, $\mu|_w = \mu$ by definition:*

$$\begin{aligned}\mu|_w(vX^{\mathbf{N}}) &= \frac{\mu(wvX^{\mathbf{N}})}{\mu(wX^{\mathbf{N}})} \\ &= \frac{\prod_{i=0}^{|w|-1} p(w_i) \prod_{j=0}^{|v|-1} p(v_j)}{\prod_{i=0}^{|w|-1} p(w(i))} \\ &= \prod_{j=0}^{|v|-1} p(v_j) = \mu(vX^{\mathbf{N}}).\end{aligned}$$

  Another way to say it is that it measures probability of independent events, where the next outcome does not depend on the previous ones.

- *a Markov measure μ has at most $|X| + 1$ nontrivial sections: μ and $\mu|_x$ for $x \in X$. This is because $\mu|_{wx} = \mu|_x$ for all admissible words $w \in X^*$. Indeed, assuming w is not the empty word (since otherwise there's nothing to show),*

$$
\begin{aligned}
\mu|_{wa}(vX^{\mathbf{N}}) &= \frac{\mu(wavX^{\mathbf{N}})}{\mu(waX^{\mathbf{N}})} \\
&= \frac{\left(l(w_0)\prod_{i=1}^{|w|-1} L(w_{i-1},w_i)\right) L(w_{|w|-1},a) \left(L(a,v_0)\prod_{j=1}^{|v|-1} L(v_{j-1},v_j)\right)}{\left(l(w_0)\prod_{i=1}^{|w|-1} L(w_{i-1},w_i)\right) L(w_{|w|-1},a)} \\
&= L(a,v_0)\prod_{j=1}^{|v|-1} L(v_{j-1},v_j) \\
&= \frac{l(a)L(a,v_0)\prod_{j=1}^{|v|-1} L(v_{j-1},v_j)}{l(a)} \\
&= \frac{\mu(avX^{\mathbf{N}})}{\mu(aX^{\mathbf{N}})} \\
&= \mu|_a(vX^{\mathbf{N}}).
\end{aligned}
$$

$\triangle$

Definition 3.1.3: a k**-step Markov measure** is a measure μ such that for all words admissible $w \in X*$ of length k and all admissible $v \in X^*$, $\mu|_{vw} = \mu|_w$.

Informally, this measures the probability of events where the probability of an outcome may depend on what the preceding k outcomes was. It can be shown that k-step Markov measures are Gibbs; refer to [6] for details.

Remark 3.1.4. *A Markov measure is a 1-step Markov measure. A k-step Markov measure on $X^{\mathbf{N}}$ with $|X| = d$ is finite-state with at most $\dfrac{d^{k+1}-1}{d-1}$ sections.*

Proof. note that a finite d-tree of depth $k+1$ has $1 + d + d^2 + \ldots + d^k = \frac{d^{k+1}-1}{d-1}$ nodes, which encode all words of length not exceeding k. By definition, every nontrivial section of a k-step Markov measure is a section by one of these words. $\square$

With a finite-state measure μ, one can associate an automaton A_μ that computes the measure; this will be defined more precisely in the following proposition. We say that A_μ

determines the measure μ.

Proposition 3.1.5. *Let μ be a finite-state probability measure, with sections $\mu_1, \ldots, \mu_n$ (where $\mu_i = \mu|_{w_i}$ for some $w_i \in X^*$). Consider an automaton A_μ with input alphabet X, output alphabet $Y \subset [0,1]$, state set $S = \{\mu_1, \ldots, \mu_n\}$, initial state $s_0 = \mu \in S$, and transition and output functions defined by*

$$\pi(\mu_i, a) := \mu_i|_a; \tag{3.1}$$
$$\lambda(\mu_i, a) := \mu_i(aX^{\mathbf{N}}).$$

The automaton defined in 3.1 uniquely determines μ via the following relation: for $w \in X^*$, $A_\mu(w) = p_0 p_1 \ldots p_{|w|-1}$ is a sequence of real numbers whose product is $\mu(wX^{\mathbf{N}})$:

$$\mu(wX^{\mathbf{N}}) = \prod_{i=0}^{|w|-1} (A_\mu(w))_i. \tag{3.2}$$

Proof. The proposition holds for when $|w| = 1$ by construction; assume it holds for all words of length k. Then for $w = w_0 w_1 \ldots w_k$,

$$\begin{aligned}
\prod_{i=0}^{k} p_i &= \mu(w_0 w_1 \ldots w_{k-1} X^{\mathbf{N}}) \cdot ((\ldots(\mu|_{w_0})|_{w_1})|_{w_2})\ldots)|_{w_{k-1}}(w_k X^{\mathbf{N}}) \\
&= \mu(w_0 w_1 \ldots w_{k-1} X^{\mathbf{N}}) \mu|_{w_0 w_1 \ldots w_{k-1}}(w_k X^{\mathbf{N}}) \text{ by inductively applying Prop. 2.2.7} \\
&= \mu(w_0 w_1 \ldots w_k X^{\mathbf{N}}) = \mu(wX^{\mathbf{N}}).
\end{aligned}$$

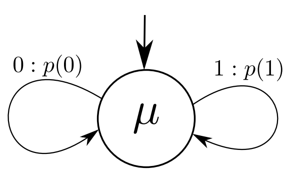
The proposition holds by induction. □

Definition 3.1.6: When μ and M are as in 3.1.5, wecall a state s of M **trivial** if it determines a trivial section if μ. Refer to Example 3.1.12 for an automaton with a trivial state; in particular, the state $\mu|_1 1$ in Figure 3.4a is trivial.

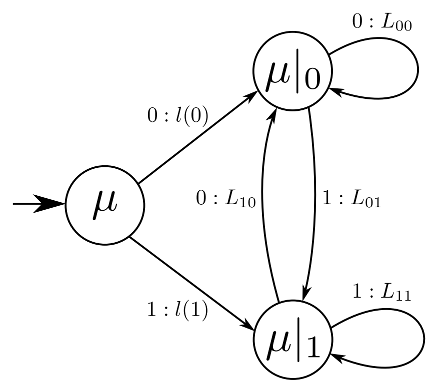
Example 3.1.7. *The automaton computing a Bernoulli measure on* $\{0,1\}^{\mathbf{N}}$ *defined by* $p = (p(0), p(1))$ *is depicted in Figure 3.1a.* $\triangle$

Example 3.1.8. *The automaton computing a Markov measure on* $\{0,1\}^{\mathbf{N}}$ *defined by a vector* $l = (l(0), l(1))$ *and a matrix* $L = L_{ij}$ *is depicted in Figure 3.1b.* $\triangle$

Example 3.1.9. *The automaton computing a* 2*-step Markov measure on* $\{0,1\}^{\mathbf{N}}$ *defined by* $l = (l(0), l(1))$ *and* $L = L_{ij}$ *is depicted in Figure 3.2.* $\triangle$



(a) Diagram of the automaton computing a Bernoulli measure



(b) Diagram of the automaton computing a Markov measure

Figure 3.1: Automata determining a Bernoulli and a Markov measure on $\{0,1\}^{\mathbf{N}}$

Note that for a finite word w, $\mu|_w$ can be seen as a function on X by giving the measure of the cylinder $xX^{\mathbf{N}}$ for $x \in X$.

Similarly to tree automorphisms, we define the **portrait of the measure** μ to be the diagram consisting of the marked tree $\mathcal{T}$, where the node corresponding to a word w is marked with the values $\mu|_w$ takes on X. A portrait defines a measure uniquely.

When dealing with probability measures, it is often convenient to consider the vector $p_w := (\mu\_w(x_0 X^{\mathbf{N}}), \ldots, \mu|_w(x_{d-1}X^{\mathbf{N}})$ up to scaling, since $\sum_{i=0}^{d-1} \mu(x_i X^{\mathbf{N}}) = 1$, and so

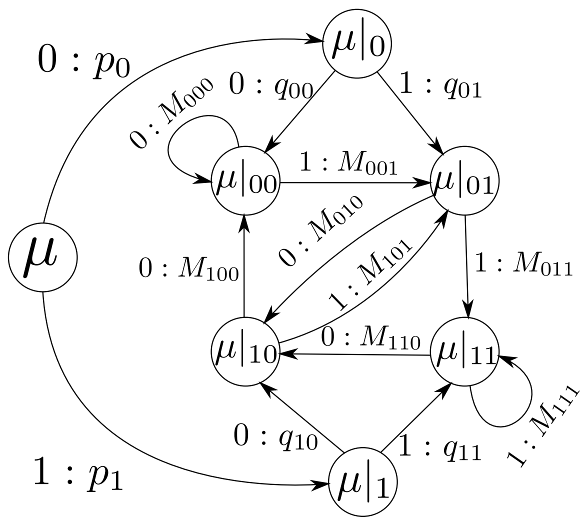


Figure 3.2: Diagram of the automaton defining a general 2-step Markov measure on $\{0,1\}^{\mathbf{N}}$

the **proportion** $p_{w_0} : p_{w_1} : \ldots : p_{w_{n-1}} \in \mathbf{R}P^n$ defines the values of μ on X unambiguously. We then use the proportion as the corresponding label in the portrait.

Example 3.1.10. *The uniform Bernoulli measure on a binary alphabet has one section with proportion* $1 : 1$*. Its portrait is shown in Figure 3.3.*

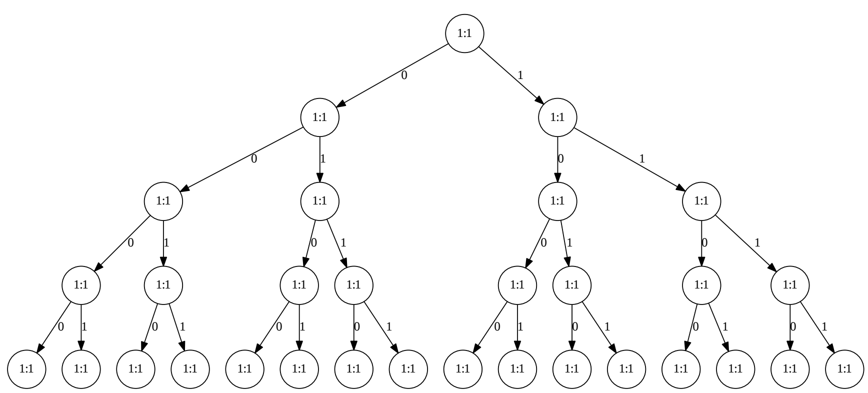


Figure 3.3: Portrait of the uniform Bernoulli measure up to level 5

Remark 3.1.11. *As with automorphisms, one can draw the portrait of any probability measure on the space $X^{\mathbf{N}}$, but not all probability measures are finite-state.*

It should be noted that even small automata define interesting finite-state measures.

Example 3.1.12. *The measure μ defined by the automaton in Figure 3.4a is a 2-step Markov measure on $\Omega = \{0,1\}^{\mathbf{N}}$ that is not a 1-step Markov measure on Ω. It is supported on the* ***Fibonacci subshift****, which is the (shift-invariant) subset of Ω consisting of all sequences that do not contain consequtive 1's. The number of nontrivial sections of μ by words of length n is the $n+1$'st Fibonacci number, as can be seen in the portrait of μ shown in Figure 3.4b.*

In drawing the portrait, we omit the subtrees corresponding to the null measure for clarity.

Example 3.1.12 necessitates the following definition:

Definition 3.1.13: we say that a state s of an automaton defining a finite-state measure is **trivial** if $\lambda(s,x) = 0$ and $\pi(s,x) = s$ for all $x \in X$. Note that s defines the trivial (i.e. null) measure.

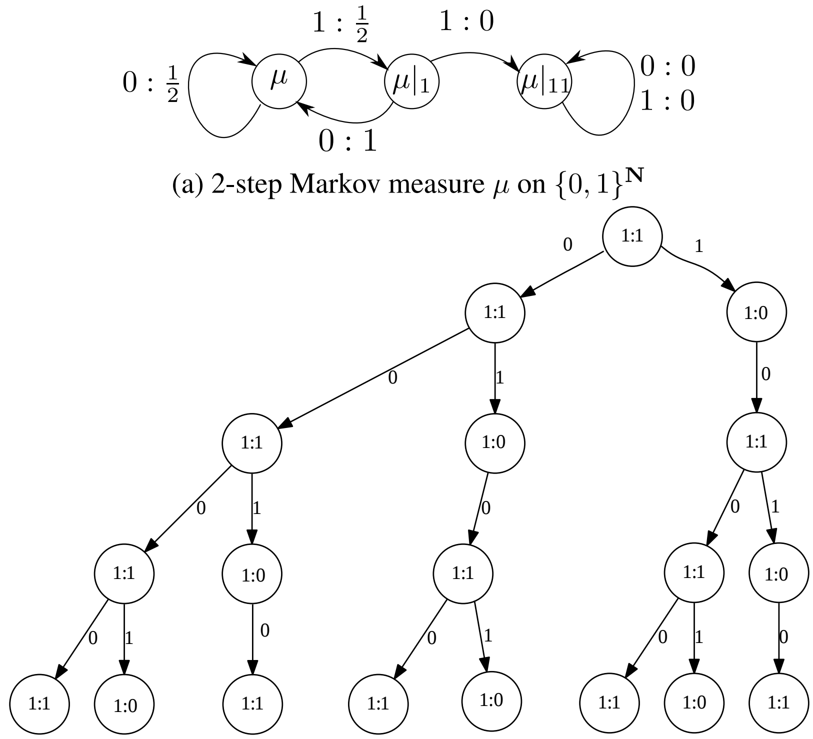
As an example, $\mu|_1 1$ in Figure 3.4a is a trivial state.

Remark 3.1.14. *Given a finite-state measure μ, the automaton defined in 3.1 is minimal and contains at most one trivial state.*

3.2 Images of finite-state measures under tree automorphisms

The following proposition is useful for constructing the automata of finite-state measures which are images of automata actions:

Proposition 3.2.1. *Let $A = (X, S, s_0, \pi, \lambda)$ be a Mealy automaton with initial state $s_0 = g$*



(a) 2-step Markov measure μ on $\{0,1\}^{\mathbf{N}}$

(b) Portrait of μ up to level 5

Figure 3.4: A finite-state measure supported on the Fibonacci subshift

acting on $\mathcal{T}$, and let ν be a probability measure on $\partial\mathcal{T}$. Then for $x \in X$,

$$(g_*\nu)(xX^{\mathbf{N}}) = \sum_{y\in\lambda_g^{-1}(x)} \nu(yX^{\mathbf{N}});$$

$$(g_*\nu)|_x = \frac{\displaystyle\sum_{y\in\lambda_g^{-1}(x)} \nu(yX^{\mathbf{N}})(g|_y)_*(\nu|_y)}{\displaystyle\sum_{y\in\lambda_g^{-1}(x)} \nu(yX^{\mathbf{N}})}. \qquad \textit{Note: } g|_y = \pi_g(y)$$

where $\pi_g(x) := \pi(g,x)$ and $\lambda_g(x) := \lambda(g,x)$.

Proof. Note that for a word $w \in X^*$,

$$g^{-1}(xwX^{\mathbf{N}}) = \bigsqcup_{y \in \lambda_g^{-1}(x)} y\pi_g(y)^{-1}(wX^{\mathbf{N}}).$$

By definition,

$$\begin{aligned}(g_*\nu)|_x(wX^{\mathbf{N}}) &= \frac{(g_*\nu)(xwX^{\mathbf{N}}}{(g_*\nu)(xX^{\mathbf{N}})} \\ &= \frac{\nu(g^{-1}(xwX^{\mathbf{N}})}{\nu(g^{-1}(xX^{\mathbf{N}}))} \\ &= \frac{\sum_{y \in \lambda_g^{-1}(x)} \nu(yX^{\mathbf{N}})\nu|_y\left(\pi_g(y)^{-1}(wX^{\mathbf{N}})\right)}{\sum_{y \in \lambda_g^{-1}(x)} \nu(yX^{\mathbf{N}})\nu|_y\left(\pi_g(y)^{-1}(X^{\mathbf{N}})\right)} \\ &= \frac{\sum_{y \in \lambda_g^{-1}(x)} \nu(yX^{\mathbf{N}})\pi_g(y)_*(\nu|_y)(wX^{\mathbf{N}})}{\sum_{y \in \lambda_g^{-1}(x)} \nu(yX^{\mathbf{N}})}. \square\end{aligned}$$

Corollary 3.2.2. *When g is as in Prop. 3.2.1, and ν is a Bernoulli measure given by probability vector p,*

$$(g_*\nu)|_x = \frac{\sum_{y \in \lambda_g^{-1}(x)} p(y)\pi_g(y)_*(\nu)}{\sum_{y \in \lambda_g^{-1}(x)} p(y)}.$$

In particular, when ν is uniform Bernoulli, $(g_*\nu)(xX^{\mathbf{N}}) = |\lambda_g^{-1}(x)|/|X|$, and

$$(g_*\nu)|_x = \frac{1}{|\lambda_g^{-1}(x)|} \sum_{y \in \lambda_g^{-1}(x)} \pi_g(y)_*(\nu).$$

When ν is uniform Bernoulli, its pushforwards by invertible endomorphisms are easy:

Proposition 3.2.3. *When ν is uniform Bernoulli and g is invertible, $g_*\nu = \nu$.*

Proof. For $w \in X^*$,

$$g_*\nu(wX^{\mathbf{N}}) = \nu(g^{-1}(wX^{\mathbf{N}}) = \nu(g^{-1}(w)X^{\mathbf{N}}) = |X|^{-|w|} = \nu(wX^{\mathbf{N}}). \square$$

We now state a consequence of 3.2.1 on the sections of images of measures for invertible automata:

Corollary 3.2.4. *Let g be a state of an invertible Mealy automaton A, let μ be a probability measure on $X^{\mathbf{N}}$, and $x \in X$. Let $y = g^{-1}(x)$, and $h = \pi(g, y)$. Then*

$$(g_*\mu)(xX^{\mathbf{N}}) = \mu(yX^{\mathbf{N}});$$
$$(g_*\mu)|_x = h_*(\mu|_y).$$

A consequence of this is the following theorem:

Theorem 3.2.5. *Finite-state measures are preserved by action of invertible finite automata.*

That is, if g is the initial state of an invertible finite automaton A, and μ is a finite-state measure, then $g_*\mu$ is also a finite-state measure.

Proof: Let M be the automaton that computes μ. We construct a finite automaton N that computes $g_*\mu$.

Let $A = (X, X, S_A, g, \pi_A, \lambda_A)$, and $M = (X, Y, S_M, \mu, \pi_M, \lambda_M)$ (here, $Y \subset [0, 1]$, $\mu \in S_M$, and S_M is the finite set of sections of μ).

Let $S_N := \{s_*\nu \ : \ s \in S_A, \nu \in S_M\}$.

Consider $N = (X, Y, S_N, g_*\mu, \pi_N, \lambda_N)$. Let $s \in S_A, \nu \in S_M$. We define π_N, λ_N using

Corollary 3.2.4:

$$\lambda_N(s_*\nu, x) := (s_*\nu)(xX^{\mathbf{N}});$$

$$\pi_N(s_*\nu, x) := (s_*\nu)|_x \in S_N \text{ by Corollary 3.2.4.}$$

This is the automaton that computes $g_*\mu$ by definition.

More explicitly, let $y = s^{-1}(x)$ (i.e. the unique character in X such that $s(y) = x$), and let $t = \pi_A(s, y)$. Then by Corollary 3.2.4:

$$\lambda_N(s_*\nu, x) = \nu(yX^{\mathbf{N}}) = \lambda_M(\nu, y);$$

$$\pi_N(s_*\nu, x) = t_*(\nu|_y) = t_*\pi_M(\nu, y).$$

This allows one to construct the automaton N when A and M are given. □

We illustrate this theorem with the following example:

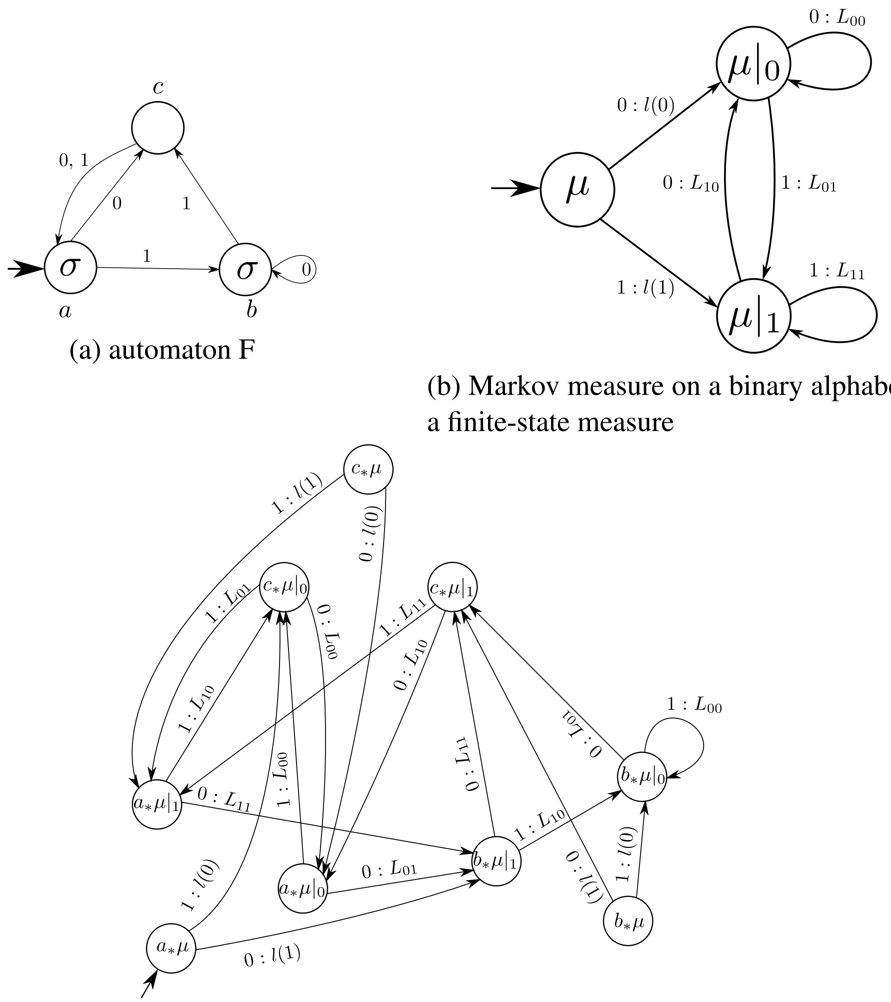
Example 3.2.6. *Let automaton F be given in Figure 3.5a, and μ be a Markov measure over a binary alphabet given by a stochastic matrix*

$$L = \begin{bmatrix} L_{00} & L_{01} \\ L_{10} & L_{11} \end{bmatrix} \tag{3.3}$$

and a stationary vector $l = (l(0), l(1))$. We represent μ as a finite-state measure with the automaton in Figure 3.5b. The automaton that computes $a_*\mu$ is presented in Figure 3.5c.

Theorem 3.2.5 explicitly specifies the transition and output functions for the automaton $g_*\mu$. However, it can be formulated in a much shorter way.

Theorem 3.2.7. *Let A be an invertible finite automaton with initial state g, and μ be a finite-state measure with automaton M.*



(a) automaton F

(b) Markov measure on a binary alphabet as a finite-state measure

(c) Automaton for the finite-state measure $\mathbf{F}_*\mu$

Figure 3.5: An automaton, a finite-state measure, and its image

Then the automaton for $g_*\mu$ is $M \cdot A^{-1}$, where $\cdot$ is the automaton product. For $s \in S_A, \nu \in S_M$, $s_*\nu$ is computed by the state (ν, s).

Proof. this follows directly from the definition of product automaton and Theorem 3.2.5.

Another way to see the same is that for $w \in X^*$,

$$g_*\mu(wX^{\mathbf{N}}) = \mu(g^{-1}(wX^{\mathbf{N}})) = \mu(g^{-1}(w)X^{\mathbf{N}}).$$

Therefore, the automaton $M \cdot A^{-1}$ computes $g_*\mu$ by the definition of the automaton of a finite state measure. $\square$

3.3 Conditions for the image a finite-state measure to be finite-state

In this section, we expand the results of Section 3.2 to study the image $g_*\mu$ of a finite-state measure probability measure μ when g is given by a non-invertible automaton.

Observation: from Proposition 3.2.1, we see that a section of $g_*\mu$ by $x \in X$ is a linear combinations of images of sections of μ by sections of g. Combining this with the result of Corollary 2.2.8, we obtain the following

Proposition 3.3.1. *Let g and μ be finite-state, with sections $g_1, \ldots, g_k$ and $\mu_1, \ldots, \mu_m$, respectively.*

Then the sections of $g_*\mu$ lie in the km-dimensional vector space $\mathcal{V}_{g,\mu}$ spanned by $\mathcal{B}_{g,\mu} = \{g_{i*}\mu_j \ : \ 1 \leq i \leq k, 1 \leq j \leq m\}$.

Proof. Let $w \in X^*$. The result holds when $|w| = 1$ as a direct consequence of Proposition 3.2.1 and Corollary 2.2.8. The result then holds by induction on $|w|$. $\square$

We can therefore write a section of $g_*\mu$ as a row vector in the basis $\mathcal{B}_{g,\mu}$:

$$r = (r_{11}, \ldots, r_{km}) = \sum i = 1^k \sum_{j=1}^{m} r_{ij} g_{i*}\mu_j.$$

Since a probability measure ν and its nontrivial sections satisfy $\nu(X^{\mathbf{N}}) = 1$, only one vector in the span of r defines a probability measure. Therefore, it makes sense to consider r as an element of the projectivized space $P\mathcal{V}_{g,\mu}$ (formally defined below), writing it as

$$[r] = [r_{11} : \ldots : r_{km}].$$

Definition 3.3.2: When μ is a finite-state measure, and g is a finite-state endomoprhism of $\mathcal{T}$,

$$P\mathcal{V}_{g,\mu} = \mathcal{V}_{g,\mu}/(\nu \sim \alpha\nu, \alpha \in \mathbf{R}),$$

where $\mathcal{V}_{g,\mu}$ us as in Proposition 3.3.1.

Theorem 3.3.3. *Let μ be a finite-state measure given by an automaton $M = (S, \mu, X, \pi, \lambda)$, g be a finite-state endomorphism of $\mathcal{T}$.*

Then X acts on $P\mathcal{V}_{g,\mu}$ (Def. 3.3) linearly by sections:

$$x \cdot \mu := \mu|_x.$$

The action is represented by a matrix $\mathcal{M}_x$ given by

$$\mathcal{M}_x(s_*\nu, t_*\xi) = \sum_{\substack{y \in X \\ s(y)=x \\ s|_y = t \\ \nu|_y = \xi}} \lambda(\nu, y)$$

where s, t are sections of g, and ν, ξ are sections of μ, and λ is the output function in the automaton corresponding to μ. Then when μ is represented by a row projective vector $[r]$, $\mu|_x$ is represented by $[r \cdot \mathcal{M}_x]$.

Proof. This is a direct consequence of the observation above; we simply apply the formula in Proposition 3.2.1 and Corollary 2.2.8 and ignore the normalizing factor $\dfrac{1}{\mu(xX^{\mathbf{N}})}$:

$$\mathcal{M}_x(s_*\nu, t_*\xi) = s_*\nu(xX^{\mathbf{N}}) \sum_{\substack{y\in X \\ s(y)=x \\ s|_y=t \\ \nu|_y=\xi}} \nu(yX^{\mathbf{N}}) \frac{1}{s_*\nu(xX^{\mathbf{N}})}$$

$$= \sum_{\substack{y\in X \\ s(y)=x \\ s|_y=t \\ \nu|_y=\xi}} \lambda(\nu, y). \ \square$$

We now obtain the criterion for the image being finite-state.

Theorem 3.3.4. *Let g be a finite-state automorphism, μ a finite state measure, and let $[\boldsymbol{v}]$ be the row vector representing $[g_*\mu] \in \mathcal{PV}_{g,\mu}$ (using the basis $\mathcal{B}_{g,\mu}$). Let $\mathcal{O}$ be the orbit of $\boldsymbol{v}$ under the action of the free semigroup generated by the matrices $\mathcal{M}_x$ for $x \in X$:*

$$\mathcal{O} = \left\{ \boldsymbol{v} \cdot \prod_{i=1}^{n} \mathcal{M}_{x_i} \ : \ n \in \mathbf{N} \cup 0, x_i \in X \right\}.$$

Then points in $\mathcal{O}$ are representations of the sections of $g_*\mu$.

In particular, $g_*\mu$ is finite-state if and only if $\mathcal{O}$ is finite.

Proof. by Theorem 3.3.3, for $w = w_1 w_1 \dots w_n \in X^*$, $g_*\mu|_w$ is represented by

$$\mathbf{v} \cdot \prod_{i=1}^{n} \mathcal{M}_{w_i}.$$

The result follows. $\square$

We develop examples for Theorem 3.3.4 in Section 4.3 with μ being the uniform Bernoulli measure. In particular, Example 4.3.7 illustrates the case when the orbit $\mathcal{O}$ is finite, and Example 4.3.8 illustrates the case when $\mathcal{O}$ is infinite.

3.3.1 Quasi-finite-state measures

Observations in Section 3.3 necessitate the following definition:

Definition 3.3.5: a measure μ is called **quasi-finite-state** if the dimension of the vector space spanned by its sections is finite.

Corollary 3.3.6. *Quasi-finite-state measures are preserved by actions of tree endomorphisms. In particular, images of (quasi)finite-state measures under such actions are quasi-finite-state.*

Proof. This is immediate from Theorem 3.3.4. □

Note that the so-called 1**-block maps** (maps $X^{\mathbf{N}} \to Y^{\mathbf{N}}$ induced by a surjective map $\pi : X \to Y$) are given by non-invertible automata. Without loss of generality, one may consider Y as a proper subset of X. **Sofic** measures are images of Markov measures under 1-block maps. We obtain the following:

Proposition 3.3.7. *Sofic measures are quasi-finite-state.*

3.4 Conditions for a finite-state measure to be k-step Markov

The following theorem (whose proof was communicated by Y. Vorobets) provides necessary and sufficient conditions for a finite-state measure to be k-step Markov:

Theorem 3.4.1. *Let μ be a finite-state measure, and let M be the automaton that computes it. Assume M is minimal and strongly-connected, and $\mu_1, \dots, \mu_n$ are its nontrivial states.*

Then μ is k-step Markov (for some k) if and only if for any word $w \in X^*$, there is at most one i, $1 \le i \le n$, such that $\mu_i|_w = \mu_i$.

When μ is k-step Markov, $k \le n(n-1)+1$.

Proof. ⇒ Let μ be k-step Markov. Assume, for a contradiction, that the hypothesis of the theorem does not hold, and there exist two states, s and t, and a word w such that $\pi(s,w) = s$ and $\pi(t,w) = t$.

Let s_0 be the initial state of M. Let w_s be a word such that $\pi(s_0, w_s) = s$, and let w_t be a word such that $\pi(s_0, w_t) = t$; such words exist because M is strongly-connected by assumption.

Fix $u = www \dots w$ such that $|u| > k$. Then

$$\pi(s_0, w_s u) = \pi(s, u) = \pi(s, ww \dots w) = s$$
$$\pi(s_0, w_t u) = \pi(t, u) = \pi(t, ww \dots w) = t$$

Now the sections $\mu|_{w_s u} = s$ and $\mu|_{w_t u} = t$ are different, but $|w_s u| = |w_t u| > k$, i.e. the section cannot be determined by examining the suffix of length k (see definition 3.1). This contradicts the assumption that μ is k-Markov. $\square$

$\Leftarrow$: assume the hypothesis holds. For μ to be k-step Markov, it suffices to show that for any two distinct, nontrivial states s and t, and any word w with $|w| = k$, either $\pi(s, w) = \pi(t, w)$ if neither is a trivial state.

Let $k = n(n-1)$, and fix w with $|w| = k$. Consider the list of pairs

$$(s, t)$$
$$(\pi(s, w_0), \pi(s, w_1))$$
$$(\pi(s, w_0 w_1), \pi(t, w_0 w_1))$$
$$\dots$$
$$(\pi(s, w), \pi(t, w))$$

Assume that this list doesn't contain trivial states, the definition of k-step Markov measure only applies to admissible words.

If any pair contains two non-distinct states, we are done: if $\pi(s, w_0 w_1 \dots w_i) =$

$\pi(t, w_0 w_1 \dots w_i) = q$ for some $i, 0 \le i < k$, then $\pi(s, w) = \pi(t, w) = \pi(q, w_{i+1} \dots w_{k-1}$.

Therefore we assume that it is not the case, and all pairs in the list contain two distinct states.

Since there are n nontrivial states by assumption, there are at most $n(n-1)$ distinct pairs of nontrivial states. Since $k > n(n-1)$, there is a repetition in this list, i.e.

$$(\pi(s, w_0..w_i), \pi(t, w_0..w_i)) = (\pi(s, w_0..w_j), \pi(t, w_0..w_j))$$

for some $0 \le i < j \le k$. But that means that the word $w' = w_{i+1} \dots w_j$ fixes two distinct states $s' = \pi(s, w_0..w_i)$ and $t' = \pi(t, w_0..w_i)$; that is, $\pi(s', w') = \pi(t', w')$ and $s' \ne t'$. This contradicts the hypothesis. The theorem holds by contradiction. $\square$

Remark: the free semigroup $FS(X)$ generated by X acts on the states of an automaton M: for $w \in FS(X)$, $w \cdot s := \pi(s, w)$. The condition of Theorem 3.4.1 can be re-stated as follows: for all $w \in X^*$, the action of w on the states of M has either no fixed points, or only one fixed point. This motivates the following:

Definition 3.4.2: an automaton is **unifixed** if it satisfies the condition of Theorem 3.4.1.

Corollary 3.4.3. *Let g be a tree automorphism given by automaton A. The image $g_* \mu$ of a Markov measure μ is k-step Markov (and hence Gibbs) if and only if the automaton $M \cdot A^{-1}$ is unifixed.*

Remark 3.4.4. *Not all minimal, strongly-connected automata are unifxied. The automaton is in Figre 3.6 is an example of such automaton for which the condition of Theorem 3.4.1 fails: the action of the word* 01 *on the states of the automaton has two fixed points (the two upper states in the diagram). Therefore, the finite-state measure defined by this automaton is not a k-step-Markov measure.*

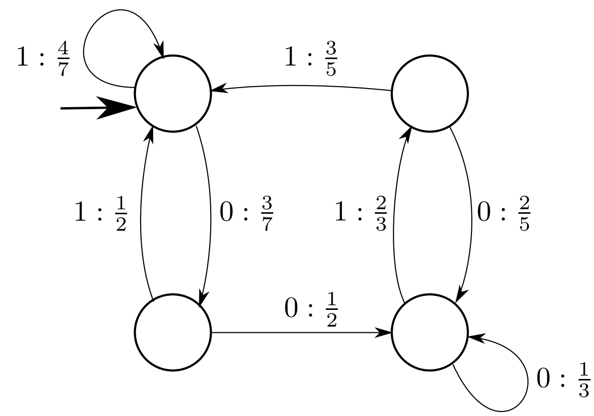


Figure 3.6: A diagarm of the automaton of a finite-state measure that is not k-step Markov for any k.

.

3.5 When finite-state measures are not Gibbsian

Recall definition 2.2.1 of a Gibbs measure.

Since k-step Markov measures are Gibbs, Corollary 3.4.3 allows us to obtain a sufficient condition for images of Markov mesaures to be Gibbs.

Example 3.5.1. *Let a be the tree automorphism given by $automatonF$, and let μ be a Markov measure on a binary alphabet (Figure 3.5a and 3.5b). The automaton N defining the measure $a_*\mu$ (Figure 3.5c) is not unifixed. Indeed,*

$$\pi_N(a_*\mu|_0, 1010) = a_*\mu|_0$$

$$\pi_N(b_*\mu|_1, 1010) = b_*\mu|_1$$

Therefore, the measure $a_*\mu$ is not k-step Markov for any k. $\triangle$

Furthermore, we show that $q_*\mu$ is not Gibbs. Informally, the idea behind the proof is that we can't tell by the tail of a word which $0101...$ cycle we're in, and the outputs of the two $0101..$ cycles can be very different. This idea is formalized in the following:

Theorem 3.5.2. *Let μ be a finite-state measure with automaton M that is not unifixed, and let $w \in X^*$ and two sections $\mu_1 = \mu|_{v_1}$ and $\mu_2 = \mu|_{v_2}$ of μ be such that $\pi(\mu_1, w) = \mu_1$ and $\pi(\mu_2, w) = \mu_2$.*

*Assume $\mu_i(v_i w X^{\mathbf{N}}) \neq 0$ for $i = 1, 2$, and $\mu_1(wX^{\mathbf{N}}) \neq \mu_2(wX^{\mathbf{N}})$. Then μ is **not** Gibbs.*

Proof: For a contradiction, assume μ is Gibbs, and let f be the potential function.

Recall that for a section s of μ,

$$s(wX^{\mathbf{N}}) = \lambda(s, w_1) \cdot \pi(s, w_1)(\sigma(w)X^{\mathbf{N}})$$

by definition of the automaton computing a finite-state measure (see 3.2). Let $p_i = \mu(v_i X^{\mathbf{N}})$, and and let $q_i = \mu_i(wX^{\mathbf{N}})$. Then by assumption of the theorem, $p_i \neq 0$, $q_i \neq 0$, and

$$\mu(v_i w^n X^{\mathbf{N}}) = p_i q_i^n.$$

Let $W = w^{\mathbf{N}} \in X^{\mathbf{N}}$. By the assumption, μ is Gibbs, so there exists $C > 0$ such that for $i = 1, 2$,

$$\frac{1}{C} < \frac{\mu(v_i w^n X^{\mathbf{N}})}{\exp\left(\sum_{k=0}^{|v_i w^n|} f(\sigma^k(v_i W))\right)} < C. \tag{3.4}$$

Let

$$b_i := \exp\left(\sum_{k=0}^{|v_i|} |f(\sigma^k(v_i W))\right).$$

We can now rewrite the inequality 3.4 as follows:

$$\frac{1}{C} < \frac{p_i q_i^n}{b_i \exp f(W)} < C$$
$$\frac{1}{C}\frac{b_i}{p_i} < \frac{q_i^n}{\exp f(W)} < C\frac{b_i}{p_i}$$

Considering the above for $i = 1, 2$, we obtain:

$$\frac{1}{C^2}\frac{b_1 p_2}{b_2 p_1} < \frac{q_1^n}{q_2^n} < C^2\frac{b_1 p_2}{b_2 p_1}$$

Without loss of generality, assume $q_1 > q_2$, and consider the limit as $n \to \infty$ of each term. We arrive at a contradiction: the lower and upper bounds in 3.5 do not depend on n, but $(q_1/q_2)^n$ grows without bound.

Therefore, there is no way to associate a Gibbs potential function f with μ, and μ is not Gibbs. □

Example 3.5.3. *The measure $a_*\mu$ of Figure 3.5c is not Gibbs when L (see 3.3) has positive entries, and $L_{00}^4 \neq L_{11}^2 L_{10} L_{01}$*

The preceding theorem motivates the following definition. Let us call a finite-state measure μ determined by an automaton M **quasi-unifixed** if for every w in the free semigroup $FS(X)$, if μ_1 and μ_2 are fixed points of the action of w on the states of M, then $\mu_1(wX^{\mathbf{N}}) = \mu_2(wX^{\mathbf{N}})$.

The proof of Theorem 3.5.2 does not apply to quasi-unifixed measures that are not unifixed. We put forth the following

Conjecture 3.5.4. *All quasi-unifixed finite-state measures are Gibbsian.*

3.6 Finite-state measures, random walks and Markov chains

With a finite-state measure μ one can associate a random walks on the diagram of the automaton M of μ. Define a random walk on the states of M with transition probability $\lambda(s, x)$ for the (directed) edge $s \to \pi(s, x)$. Then

$$\mu(wX^{\mathbf{N}}) = P\left(\mu \xrightarrow[\mu(w_1 X^{\mathbf{N}})]{w_1} \mu_1 \xrightarrow[\dots]{w_2} \dots \xrightarrow[\dots]{w_n} \mu_n\right),$$

where P stands for the probability of a path (this follows directly from definitions; cf. 3.2 in Proposition 3.1.5).

Remark: Every finite-sate measure defines a random walk. Conversely, a random walk on a directed graph with constant out-degree d gives rise to a finite-state measure under labeling the edges of the graph with elements of an alphabet X of size d as input symbols, and putting the corresponding probabilities as output.

Definition 3.6.1: A finite-state measure μ determined by the automaton $M = (S, \mu_0 = \mu, X, \pi, \lambda)$ defines a Markov chain on the set of sections of μ with initial distribution $\mathbf{p} = \mathbf{p}_\mu$ and transition probability matrix $\mathbf{P} = \mathbf{P}_\mu$ given by

$$\mathbf{p}(\mu_i) = \begin{cases} 1, \text{if } \mu_i = \mu_0 = \mu \\ 0, \text{otherwise} \end{cases} \tag{3.5}$$

$$\mathbf{P}(\mu_i, \mu_j) = \sum_{x \in X\ :\pi(\mu_i, x) = \mu_j} \lambda(\mu_i, x). \tag{3.6}$$

Example 3.6.2. *The measure $a_*\mu$ of Figure 3.5c defines a Markov chain given by the* *stochastic matrix*

$$\boldsymbol{P}_{a_*\mu} = \begin{pmatrix} 0 & 0 & 0 & p & 1-p & 0 \\ 0 & 0 & 0 & 1-q & q & 0 \\ 0 & 0 & 1-p & 0 & 0 & p \\ 0 & 0 & q & 0 & 0 & 1-q \\ 1-p & p & 0 & 0 & 0 & 0 \\ q & 1-q & 0 & 0 & 0 & 0 \end{pmatrix}$$

The Markov chain defined in 3.6 allows us to apply the rich theory of Markov chains for finite-state measures.

Proposition 3.6.3. *Let μ be a finite-state measure determined by $M = (S, s_0 = \mu, X, \pi, \lambda)$. Suppose $\boldsymbol{P}_\mu$ has a stationary probability vector p (such that $p\boldsymbol{P} = p$). Define a measure $\tilde{\mu}$ by*

$$\tilde{\mu} = \sum_{s \in S} p_s s.$$

Then $\tilde{\mu}$ is a shift-invariant measure (w.r.t. the shift operator $\sigma : X^{\mathbf{N}} \to X^{\mathbf{N}}$).

Proof: suffices to show for the cylinder sets $wX^{\mathbf{N}}$:

$$\begin{aligned}
\tilde{\mu}(\sigma^{-1}(wX^{\mathbf{N}})) &= \sum_{x\in X} \tilde{\mu}(xwX^{\mathbf{N}}) \\
&= \sum_{x\in X}\sum_{s\in S} p_s s(xwX^{\mathbf{N}}) \\
&= \sum_{x\in X}\sum_{s\in S} p_s \lambda(s,x)\pi(s,x)(wX^{\mathbf{N}}) \\
&= \sum_{t\in S}\left(\sum_{s\in S}\sum_{x\in X\,:\,\pi(s,x)=t} p_s\lambda(s,x)t(wX^{\mathbf{N}})\right) \\
&= \sum_{t\in S}\left(\sum_{s\in S} p_s\left(\sum_{x\in X\,:\,\pi(s,x)=t}\lambda(s,x)\right)\right)t(wX^{\mathbf{N}}) \\
&= \sum_{t\in S}\left(\sum_{s\in S} p_s\mathbf{P}_{s,t}\right)t(wX^{\mathbf{N}}) \\
&= \sum_{t\in S} p_t t(wX^{\mathbf{N}}) = \tilde{\mu}(wX^{\mathbf{N}}). \ \Box
\end{aligned}$$

Lemma 3.6.4. *Let μ, **P** and $\tilde{\mu}$ be as in Proposition 3.6.3. Let μ be defined by an automaton $M = (S, \mu_1, X, \pi, \lambda)$ with the set of sections $S = \{\mu_1, \ldots, \mu_l\}$, and let $\Lambda = (\mu_1, \ldots, \mu_l)$. Let $sigma : X^{\mathbf{N}} \to X^{\mathbf{N}}$ be the (one-sided) shift. Then for $1 \le t \le l$,*

$$\sigma_* \mu_t = (P\Lambda)_t.$$

Proof. on a cylinder set $w_1 \dots w_n X^{\mathbf{N}}$,

$$\begin{aligned}
\sigma_* \mu_t(w_1 \dots w_n X^{\mathbf{N}}) &= \mu(\sigma^{-1}(w_1 \dots w_n X^{\mathbf{N}})) \\
&= \sum_{x \in X} \mu_t(x w_1 \dots w_n X^{\mathbf{N}}) \\
&= \sum_{x \in X} \lambda(\mu_t, x) \cdot \pi(\mu_t, x)(w_1 \dots w_n X^{\mathbf{N}}) \\
&= \sum_{q=1}^{l} \left(\sum_{\substack{x \in X \\ \pi(\mu_t, x) = \mu_q}} \lambda(\mu_t, x) \right) \mu_q(w_1 \dots w_n X^{\mathbf{N}}) \\
&= \sum_{q=1}^{l} \mathbf{P}_{tq} \mu_q(w_1 \dots w_n X^{\mathbf{N}}) \\
&= (P\Lambda)_t(w_1 \dots w_n X^{\mathbf{N}}) \; \square
\end{aligned}$$

Proposition 3.6.5. *Let μ, M, **P**, $\tilde{\mu}$ and Λ be as in Lemma 3.6.4. Assume **P** is irreducible. Then $\tilde{\mu}$ is ergodic (with respect to the shift σ), and is mixing whenever **P** is aperiodic.*

Note: in this proof, for $s \in S$ - a state of M, we will write μ_s for the section of μ given by s.

Proof. we need to show that

$$\lim_{k \to \infty} \frac{1}{k} \sum_{i=0}^{k-1} \tilde{\mu}(A \cap \sigma^{-k} B) = \tilde{\mu}(A) \tilde{\mu}(B) \tag{3.7}$$

for all cylinder sets $A = x_1 x_2 \dots x_m X^{\mathbf{N}}$ and $B = y_1 y_2 \dots y_n X^{\mathbf{N}}$.

Since we are taking the limit as $n \to \infty$, we can consider the sum in 3.7 starting from

$k = m$ and replace the expression under the limit with

$$\frac{1}{k}\sum_{i=0}^{k-1}\tilde{\mu}(A \cap \sigma^{-k}B) = \frac{1}{k}\sum_{i=m}^{k-1}\sum_{\substack{w\in X^* \\ |w|=i+1-m}}\sum_{s\in S} p(s)\mu_s(x_1 \dots x_m X^{\mathbf{N}}) \cdot \mu_t(wy_1y_2 \dots y_n X^{\mathbf{N}}), \tag{3.8}$$

where

$$s = s_1 \xrightarrow{x_1} s_2 \xrightarrow{x_2} \dots s_n \xrightarrow{x_n} t.$$

The right-hand side of 3.8 is then equal to

$$\begin{aligned}
&= \sum_{s\in S} p(s)\mu_s(x_1 \dots x_m X^{\mathbf{N}})\frac{1}{k}\sum_{i=m}^{k-1}\sum_{\substack{w\in X^* \\ |w|=i+1-m}} \mu_t(wy_1 \dots y_n X^{\mathbf{N}}) \\
&= \sum_{s\in S} p(s)\mu_s(A)\frac{1}{k}\sum_{i=m}^{k-1}\left(\sigma_*^{i+1-m}\mu_t\right)(B) \\
&= \sum_{s\in S} p(s)\mu_s(A)\left[\left(\frac{1}{k}\sum_{i=1}^{k-m}\mathbf{P}^i\right)\Lambda\right]_t (B)
\end{aligned}$$

However, when **P** is irreducible, from the theory of Markov measures

$$\frac{1}{k}\sum_{i=1}^{k-m}\mathbf{P}^i \to \overline{P} = \begin{pmatrix} p(1) & \dots & p(l) \\ \dots & & \dots \\ p(1) & \dots & p(l) \end{pmatrix}.$$

That is, the components of $\overline{P}_{ij} = p(j)$ do not depend on the row.

Hence the RHS of 3.8 is equal in the limit to:

$$
\begin{aligned}
&= \sum_{s\in S} p(s)\mu_s(A) \begin{pmatrix} \sum_{s\in S} p(s)\mu_s \\ \dots \\ \sum_{s\in S} p(s)\mu_s \end{pmatrix}_t (B) \\
&= \sum_{s\in S} p(s)\mu_s(A) \left(\sum_{s\in S} p(s)\mu_s\right)(B) \\
&= \tilde{\mu}(B) \sum_{s\in S} p(s)\mu_s(A) \\
&= \tilde{\mu}(B)\tilde{\mu}(A). \square
\end{aligned}
$$

Corollary 3.6.6. *Assume that the automaton M defining the finite-state measure μ is strongly-connected (in particular, μ has no trivial sections). Then $\tilde{\mu}$ is ergodic.*

Proof. when the graph of M is strongly-connected, **P** is irreducible. This is a classical result (see, e.g., [7]). $\square$

Lemma 3.6.7. *Assume that μ is as before. Then $\mu \ll \tilde{\mu}$.*

Proof. We need to show that for a cylinder set $wX^{\mathbf{N}}$, $w \in X^*$, $\tilde{\mu}(wX^{\mathbf{N}}) = 0 \Rightarrow \mu(wX^{\mathbf{N}}) = 0$. But when the automaton of μ is strongly-connected, $\tilde{\mu}$ is a linear combination of sections of μ with positive weights: $\tilde{\mu} = \sum_{s\in S} p(s)\mu_s$ with $p(s) > 0$. Then

$$\tilde{\mu}(wX^{\mathbf{N}}) = 0 \Rightarrow \mu_s(wX^n n) = 0 \forall s \in S \Rightarrow \mu = \mu_{s_0} = 0. \square$$

Theorem 3.6.8. *Let μ, p be as in Propotion 3.6.5. Then the frequency of $x \in X$ w.r.t. μ is well-defined and is given by*

$$\operatorname{freq} x = \sum_{s\in S} p(s)\lambda(s, x).$$

Proof: we apply the pointwise Ergodic theorem: for $\tilde{\mu}$-almost all $w \in X^{\mathbf{N}}$,

$$\begin{aligned}\operatorname{freq}(x) = \lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \chi_x \sigma^n(w) \\ &= \int_{X^{\mathbf{N}}} \chi_x \, d\tilde{\mu} \\ &= \tilde{\mu}(xX^{\mathbf{N}}) \\ &= \sum_{s\in S} p(s)\lambda(s,x).\end{aligned}$$

The theorem holds since $\mu \ll \tilde{\mu}$. $\square$

Example 3.6.9. *The measures $a_*\mu|_0$ and $a_*\mu|_1$ of Figure 3.5c have a stationary probability vector*

$$\left(\frac{q}{3(p+q)}, \frac{p}{3(p+q)}, \frac{q}{3(p+q)}, \frac{p}{3(p+q)}, \frac{q}{3(p+q)}, \frac{p}{3(p+q)}\right),$$

and the frequency vector

$$\left(\frac{2p+q}{3(p+q)}, \frac{p+2q}{3(p+q)}\right).$$

Since both sections of $a_*\mu$ have the same frequency vector, this is the frequency vector of $a_*\mu$ as well.

4. THE AUTOMATIC LOGARITHM

In this chapter we introduce the notion of the automatic logarithm, and apply it to the problem of studying the distribution of chords in a Scherier graph of action of two automata.

4.1 Definitions

Let X be a finite alphabet, $|X| = d$, so $\mathcal{T}$ is a regular d-tree with level n of size d^n.

Let A be an endomorphism of the tree $\mathcal{T}$ acting transitively on each level.

Then for any pair of words w_1, w_2 of length n, there is a unique integer k in $0..d^n - 1$ such that $A^k(w_1) = w_2$. Furthermore, if $A^k(w_1) = A^{k'}(w_1)$ for some integers k, k', then $k \equiv k' \mod d^n$. These are basic properties of finite orbits of length d^n. This leads to the following

Definition 4.1.1: on words of length n, the **displacement** $\mathbf{d}_{A,n} : d^n \times d^n \to \mathbf{Z}/K^n\mathbf{Z}$ is the function defined by

$$\mathbf{d}_{A,n}(w_1, w_2) := [k]_{d^n} \Leftrightarrow A^k(w_1) = w_2,$$

where $[k]_{d^n} \in Z/d^n Z$ is the equivalence class mod d^n. We write $[k]$ when n is fixed.

Definition 4.1.2: for $m, n \in \mathbf{N}$, with $n \geq m$, the *natural projection* $\phi_{m,n} : \mathbf{Z}/d^n\mathbf{Z} \to \mathbf{Z}/d^m\mathbf{Z}$ is defined by $\phi_{m,n}([k]_{d^n}) := [k]_{d^m}$.

The functions $d_{A,n}$ for different values of n are compatible with each other w.r.t the natural projection:

Proposition 4.1.3. *Let* $|w_1| = |w_2| = n$ *and* $a, b \in X$. *Then*

$$\phi_{n,n+1}(\boldsymbol{d}_{A,n+1}(w_1 a, w_2 b)) = \boldsymbol{d}_{A,n}(w_1, w_2).$$

Proof: let $\mathbf{d}_{A,n}(w_1, w_2) = [k]$ (so that $A^k(w_1) = w_2$), with $k \in 0..d^n - 1$.

Let $a' = A^k|_{w_1}(a)$. Then $A^k(w_1 a) = w_2 a'$. Note that

$$\begin{aligned} A^{K^{n+k}}(w_1 a) &= A^{K^n}(w_2 a') \\ &= A^{K^n}(w_2) A^{K^n}|_w(a'). \end{aligned}$$

By Prop. 2.1.16,

$$A^{d^n}|_w(a'), A^{2d^n}|_w(a'), \ldots, A^{(k-1)d^n}|_w(a'),$$

are all distinct. Since $|X| = K$, this implies $A^{td^n}|_w(a') = b$ for some $t \in 0, 1, \ldots, d-1$. Thus $A^{td^n+k}(w_1 a) = w_2 b$, whence $d_{A,n+1}(w_1 a, w_2 b) = [k + td^n]$.

Since $\phi_{n,n+1}([k + td^n]) = [k]$, the proposition holds. $\Box$.

Let B be another tree endomorphism.

Definition 4.1.4: on words of length n, $\log_{A,n}(B) : X^n \to \mathbf{Z}/d^n\mathbf{Z}$ is a function which calculates the displacement of a word by B along the orbit of A:

$$\log_{A,n}(B)(w) := d_{A,n}(w, B(w))$$

Corollary 4.1.5. *(of Prop. 4.1.3)*

$$\phi_{n,n+1}(\log_{A,n+1}(B)(wa) = \log_{A,n}(B)(w).$$

In other words, if $|w| = n$ and $a \in X$, the displacement of wa by B along the orbit of A is either the same as displacement of w, or differs by a multiple of d^n.

Corollary 4.1.6. *for any non-negative integers m, n with $m < n$, the following diagram* *commutes:*

$$
\begin{array}{ccc}
X^n & \xrightarrow{\sigma_r^{n-m}} & X^m \\
\Big\downarrow \log_{A,n}(B) & & \Big\downarrow \log_{A,m}(B) \\
\mathbf{Z}/d^n\mathbf{Z} & \xrightarrow{\phi_{m,n}} & \mathbf{Z}/d^m\mathbf{Z}
\end{array}
$$

Proof. This follows by induction from Prop. 4.1.3.

Let $\mathbf{Z}_d$ be the inverse limit of the directed system

$$\mathbf{Z}/d^n\mathbf{Z} \xrightarrow{\phi_{m,n}} \mathbf{Z}/d^m\mathbf{Z}$$

(for $m, n \in \mathbf{N}$). We make the following observation:

Proposition 4.1.7. *There exists function $\log_A(B) : \partial\mathcal{T} \to \mathbf{Z}_d$ (where $\mathbf{Z}_d$ are the d-adic integers), which restricts to $\log_{A,n}(B)$ on level n for all n.*

Proof. Take the inverse limit of the directed systems in the commutative diagram of Corollary 4.1.6 □

Remark 4.1.8. *By identifying $\mathbf{Z}_d$ with $\partial\mathcal{T}$, we say that $\log_A(B)$ acts on $\mathcal{T}$ by endomorphisms. (Here, we identify $w \in \mathbf{Z}_d$ with its representation as an infinite d-ary string $w_0 w_1 w_2 \ldots \in \partial\mathcal{T}$).*

When d is prime, Z_d is the d-adic integers. In the rest of the paper, we deal with $d = 2$, and so identify (and use interchangeably) the dyadic numbers and infinite binary sequences (elements of $\partial\mathcal{T}$).

Remark 4.1.9. *This definition can be extended from d-regular trees to spherically holomorphic trees (defined in e.g. [8]).*

4.2 The automaton computing the Log map

In this section we construct an automaton which computes the Log map of the previous section in the case when A is bounded.

Here and afterwards, A is a bounded-activity (in the sense of Definition 2.1.4) tree automorphism acting transitively on levels, and B is a tree endomorphism. An example of such endomorphism is the odometer (also known as the adding machine), whose automaton is shown in Figure 2.2a.

Remark 4.2.1. *Any thee automorphism that acts transitively on levels is conjugate to the odometer.*

To proceed, we need a technical result:

Lemma 4.2.2. *When A, B are tree endomorphisms given by finite automata, and A is bounded and acts transitively on all levels, the set of sections*

$$S_{A,B} := \{(B|_w, A^{d(w)}|_w, A^{2^{|w|}}|_w) \ : \ w \in X^*\}$$

is finite.

Proof. Let S_B be the set of states of the automaton of B. Since $A^{d(w)}, A^{2^{|w|}}|_w \in T_A$,

$$|S_{A,B}| \leq S_B \cdot |T_A|^2,$$

where $|T_A|$ is finite by Prop 2.1.20. □

See Example 4.2.5 for an explicit computation of $S_{A,B}$.

Theorem 4.2.3. *Let A, B be as above. Consider the automaton $L = L_{A,B}$ with set of* *states $S_{A,B}$, initial state $(B, \mathbf{1}, A)$, and transition and output functions π and λ as follows:*

$$\pi((\beta,\gamma,\delta),a) := (\beta',\gamma',\delta'), \textit{where}$$
$$\beta' = \beta|_a$$
$$\gamma' = \begin{cases} \gamma|_a \textit{ if } \beta(a)=\gamma(a); \\ (\gamma\delta)|_a, \textit{ otherwise}; \end{cases}$$
$$\delta' = \delta^2|_a$$
$$\lambda((\beta,\gamma,\delta),a) := \begin{cases} 0, \textit{ if } \beta(a)=\gamma(a); \\ 1, \textit{ otherwise}. \end{cases}$$

Then the transition function is well-defined, and the automaton L outputs $d(w)$ as a dyadic integer:

$$d(w) = \sum_{i=0}^{|w|-1} D(w)_i 2^i.$$

Proof. We first show that upon reading a word w, the automaton L ends up in the state

$$\left(B|_w, A^{d(w)}|_w, A^{2^{|w|}}|_w\right) \in S_{A,B}.$$

This hypothesis holds for the empty word. We proceed by induction on $|w|$. Let $|w| = k$, and let L be in the state (β,γ,δ) after reading in w.

Assume the hypothesis holds for w of length k.

To prove the inductive hypothesis, we let $a \in X$ and show that

$$(\beta',\gamma',\delta') := \pi((\beta,\gamma,\delta),a) = \left(B|_{wa}, A^{d(wa)}|_{wa}, A^{2^{|wa|}}|_{wa}\right).$$

Indeed:

1. $\beta' = \beta_a$ by definition, and

$$\begin{aligned} B|_{wa} &= (B|_w)|_a \text{ (by 2.1.12)} \\ &= \beta_a \\ &= \beta'. \end{aligned}$$

2. Note that $A^{2^{|w|}}(w) = w$ by transitivity of A. By definition, $\delta' = \delta^2|_a = \delta|_{\delta(a)}\delta|_a$. Now

$$\begin{aligned} A^{2^{|wa|}}|_{wa} &= A^{2^{|w|+1}}|_{wa} \\ &= (A^{2^{|w|}})^2|_{wa} \\ &= A^{2^{|w|}}|_{A^{2^{|w|}}(wa)}A^{2^{|w|}}|_{wa} \text{ (by 2.1.13)} \\ &= A^{2^{|w|}}|_{A^{2^{|w|}}(w)A^{2^{|w|}}|_w(a)}(A^{2^{|w|}}|_w)|_a \\ &= A^{2^{|w|}}|_{w\delta(a)}\delta|_a \text{ (by 2.1.12, inductive assumption, and } A^{2^{|w|}} = w) \\ &= (A^{2^{|w|}}|_w)|_{\delta(a)}\delta|_a \\ &= \delta|_{\delta(a)}\delta|_a \\ &= \delta'. \end{aligned}$$

3. By definition of d, $B(w) = A^{d(w)}(w)$. Note that

$$B(wa) = B(w)B|_w(a) = A^{d(w)}(w)\beta(a);$$
$$A^{d(w)}(wa) = A^{d(w)}(w)A^{d(w)}|_w(a) = A^{d(w)}(w)\gamma(a).$$

If $\beta(a) = \gamma(a)$, then $B(wa) = A^{d(w)}(wa)$, and thus $d(wa) = d(w)$ by definition of

d. Otherwise, $d(wa) = d(w) + 2^{|w|}$ since this is the only other possibility. Therefore,

$$A^{d(wa)} = \begin{cases} A^{d(w)}, \text{ if } \beta(a) = \gamma(a); \\ A^{d(w)} A^{2^{|w|}}, \text{ otherwise.} \end{cases}$$

Now we compute:

$$\begin{aligned}
A^{d(w)}|_{wa} &= (A^{d(w)}|_w)|_a \\
&= \delta|_a; \\
A^{d(w)} A^{2^{|w|}}|_{wa} &= A^{d(w)}|_{A^{2^{|w|}}(wa)} (A^{2^{|w|}}|_w)|_a \\
&= A^{d(w)}|_{A^{2^{|w|}}(w) A^{2^{|w|}}|_w(a)} \delta|_a \\
&= A^{d(w)}|_{w\delta(a)} \delta|_a \\
&= (A^{d(w)}|_w)|_{\delta(a)} \delta|_a \\
&= \gamma|_{\delta(a)} \delta|_a \\
&= (\gamma\delta)|_a
\end{aligned}$$

Therefore

$$A^{d(wa)}|_{wa} = \begin{cases} \gamma|_a \text{ if } \beta(a) = \gamma(a); \\ (\gamma\delta)|_a \text{ otherwise.} \end{cases}$$

This matches the definition of γ', and thus $\gamma' = A^{d(wa)}|_{wa}$.

This completes the proof of the hypothesis that the automaton is in state $\left(B|_w, A^{d(w)}|_w, A^{2^{|w|}}|_w\right)$ after reading w.

In particular, we have verified that the transition function π is well-defined, since its values are always in the set $S_{A,B}$.

Furthermore, we observed that

$$d(wa) = \begin{cases} d(w), \text{ if } \beta(a) = \gamma(a); \\ d(w) + 2^{|w|} \text{ otherwise.} \end{cases}$$

From this observation and the definition of λ, it follows by induction that

$$d(w) = \sum_{i=0}^{|w|-1} L(w)_i 2^i.$$

This completes the proof of the theorem. $\square$

Proposition 4.2.4. *When X, A and B are as in Theorem 4.2.3, and, additionally, B is invertible, the automaton $L_{A,B}$ is a Moore machine (as in Definition 2.1.2).*

Proof. By assumption, A is invertible, and so is $A^{d(w)}$ for any $w \in X^*$. B is invertible by assumption. By Prop. 2.1.11, their sections $\beta = B|_w$ and $\gamma = A^{d(w)}|_w$ are invertible, and so is $\beta\gamma^{-1}$.

Now $\mathrm{Perm}(\{0,1\}) = \{bb1, \sigma\}$, so either $\beta\gamma^{-1}(x) = (x)$, or $\beta\gamma^{-1}(x) = \sigma(x)$.

In the first case, $\lambda(\beta, \gamma, \delta)(x) = 0$ for $x \in \{0, 1\}$.

Otherwise, since σ has no fixed points, $\beta(x) \neq \gamma(x)$ and $\lambda(\beta, \gamma, \delta)(x) = 1$ for $x \in \{0, 1\}$. $\square$

Example 4.2.5. *Let A be the **adding machine**, also known as the **odometer** (see Figure 2.2a) with states A and **1**. Let automaton F have states $\{a, b, c\}$ and initial state a as in Figure 2.2b). We consider $\log_A F$.*

Note that

$$A^2|_a = A|_A(a)A|_a = A,$$

since $A|_0 A|_1 = A|_1 A|_0 = A$. Therefore, $A^{2^n} = A$ for all $n \in \mathbf{N}$, and so $S_{A,B} \subset$

$\{a, b, c\} \times \{A, 1\} \times \{A\}$. *Consequently,* $|S_{A,B}| \leq 6$.

Let's compute the transition and the output function for $L_{A,B}$. *By Prop. 4.2.4,* $L_{A,B}$ *is a Moore machine (the output* $\lambda(s, x)$ *only depends on the state* s*), so we let * stand for either* 0 *or* 1 *in what follows:*

$$\lambda((a, 1, A), *) = 1 \qquad \lambda((b, 1, A), *) = 1 \qquad \lambda((c, 1, A), *) = 0$$

$$\lambda((a, A, A), *) = 0 \qquad \lambda((b, A, A), *) = 0 \qquad \lambda((c, A, A), *) = 1$$

We can use this to compute the transition function:

$$\pi((a, 1, A), 0) = (c, 1, A) \qquad \pi((b, 1, A), 0) = (b, 1, A) \qquad \pi((c, 1, A), 0) = (a, 1, A)$$

$$\pi((a, 1, A), 1) = (b, A, A) \qquad \pi((b, 1, A), 1) = (c, A, A) \qquad \pi((c, 1, A), 1) = (a, 1, A)$$

$$\pi((a, A, A), 0) = (c, 1, A) \qquad \pi((b, A, A), 0) = (b, 1, A) \qquad \pi((c, A, A), 0) = (a, A, A)$$

$$\pi((a, A, A), 1) = (b, A, A) \qquad \pi((b, A, A), 1) = (c, A, A) \qquad \pi((c, A, A), 1) = (a, A, A)$$

Since $\delta = A$ *for all* $(\beta, \gamma, \delta) \in S_{A,B}$*, we omit it and write* (β, γ) *for* (β, γ, A) *for* $D_{A,B}$*. The automaton* $L_{A,B}$ *we have computed here is in in Figure 4.1.* $\triangle$

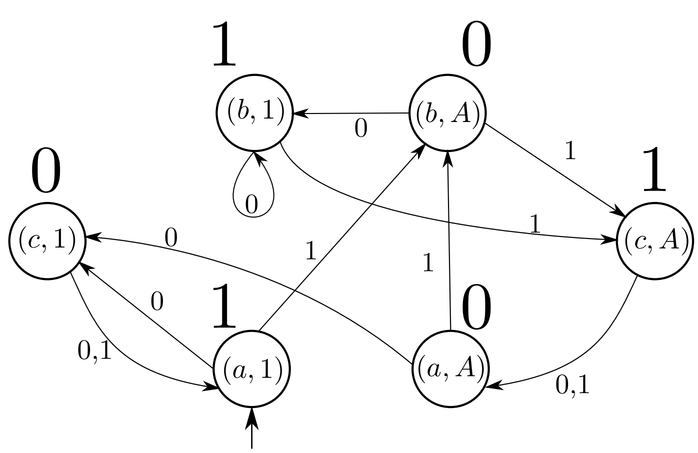


Figure 4.1: Automaton $L_{A,B}$ when A is the odometer and B is automaton F . The output from a state is the big number next to it.

	β and γ are both active or both passive	Exactly one of β and γ is active
$\pi((\beta,\gamma),a)$	$(\beta\|_a, \gamma\|_a)$	$(\beta\|_a, (\gamma A)\|_a)$
$\lambda((\beta,\gamma))$	0	1

Table 4.1: Transition and output functions of the automaton computing $\log_A(B)$ when A is the adding machine and B is invertible

Example 4.2.5 calls for a more efficient notation in the case when A is the adding machine and B is invertible:

Corollary 4.2.6. *Let A be the* ***adding machine*** *(the automaton shown in Figure 2.2a), and assume B is invertible. Then $\delta = A$ for all (β, γ, δ) in the connected component of $(B, \mathbf{1}, A)$ in $L_{A,B}$, and so can be omitted. After relabeling $(\beta, \gamma, \delta) \to (\beta, \gamma)$ in $L_{A,B}$, we obtain the Moore machine $\hat{L}_{A,B}$ with initial state $(B, \mathbf{1})$, and transition and output functions π and λ as specified in Table 4.1.*

Note: $\hat{L}$ and L are, up to relabeling, the same automaton.

Proof. Observe that

$$A^2|_a = A|_{A(a)}A|_a = A,$$

since $A|_0A|_1 = A|_1A|_0 = A$. Since the initial state is $(B, \mathbf{1}, A)$, it follows that the rest of the states in the connected component of $L_{A,B}$ containing the initial state are of the form (β, γ, A). Similarly, $\gamma \in \{\mathbf{1}, A\}$.

The rest follows from the construction 4.2.3 and Prop. 4.2.4. Note that note that $\beta(x) = \gamma(x)$ for $x \in X = \{0, 1\}$ if and only if when β and γ are both active or both passive. $\square$

Remark: in this case, one can see whether β, γ are active on the diagram of the automatons B and A, respectively.

Remark 4.2.7. *When B is invertible, and $\beta \in S(B)$ is a state of B, the transition function* *of B at β, $\lambda_\beta \in \mathrm{Perm}(X) = \{\mathbf{1}, \sigma\}$. The Table 4.1 of Prop. 4.2.6 can be written out explicitly: see Table 4.2.*

λ_β	γ	x	$\pi((\beta,\gamma),x)$	$\lambda((\beta,\gamma),x)$
$\mathbf{1}$	$\mathbf{1}$	0	$(\pi(\beta,0),\mathbf{1})$	0
$\mathbf{1}$	$\mathbf{1}$	1	$(\pi(\beta,1),\mathbf{1})$	0
σ	A	0	$(\pi(\beta,0),\mathbf{1})$	0
σ	A	1	$(\pi(\beta,1),A)$	0
$\mathbf{1}$	A	0	$(\pi(\beta,0),A)$	1
$\mathbf{1}$	A	1	$(\pi(\beta,1),A)$	1
σ	$\mathbf{1}$	0	$(\pi(\beta,0),\mathbf{1})$	1
σ	$\mathbf{1}$	1	$(\pi(\beta,1),A)$	1

Table 4.2: Table 4.1 written out explicitly

Example 4.2.8. *We compute the distance automaton when A is the odometer, and B is the Bellaterra automaton (Figure 4.2). Using the new notation:*

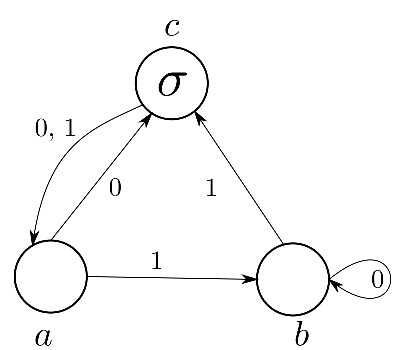


Figure 4.2: Bellaterra automaton

$$\lambda((a,1)) = 0 \qquad \lambda((b,1)) = 0 \qquad \lambda((c,1)) = 1$$

$$\lambda((a,A)) = 1 \qquad \lambda((b,A)) = 1 \qquad \lambda((c,A)) = 0$$

$$\pi((a,1),0) = (c,1) \qquad \pi((b,1),0) = (b,1) \qquad \pi((c,1),0) = (a,1)$$

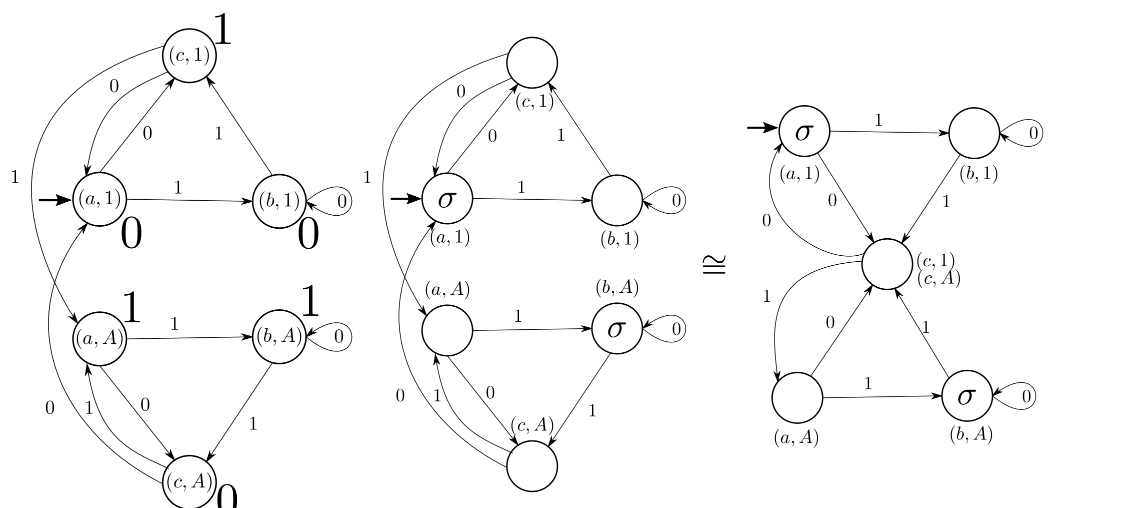
$$\pi((a,1),1) = (b,1) \qquad \pi((b,1),1) = (c,1) \qquad \pi((c,1),1) = (a,A)$$

$$\pi((a,A),0) = (c,A) \qquad \pi((b,A),0) = (b,A) \qquad \pi((c,A),0) = (a,1)$$

$$\pi((a,A),1) = (b,A) \qquad \pi((b,A),1) = (c,A) \qquad \pi((c,A),1) = (a,A)$$

△

We obtain the automaton $\tilde{L}$ in Figure 4.3a).



(a) $L_{A,B}$

(b) The delayed automaton $\sigma L_{A,B}$ is invertible

Figure 4.3: Constructions for the case of A-adding machine, B-Bellaterra

4.3 Distribution of lengths of chords

The measure we are interested in is $\mu = \mu_{\mathcal{A},\mathcal{B}} := \log_{\mathcal{A}}(\mathcal{B})_* \nu$, where ν is the uniform Bernoulli measure on $\mathcal{T}$.

The measure gives the distribution of the displacement function: for d - a finite dyadic integer written in binary as $w = w_0 \dots w_{n-1}$ (and thus interpreted as an element of $\mathbf{Z}/2^n\mathbf{Z}$),

$$\mu(wX^{\mathbf{N}}) = |\{v \in X^n \ : \ \log_{A,n}(B)(v) = d\}|.$$

Figure 2.4 illustrates the graphs of action with the cycle generated by the adding machine $\mathcal{A}$ put on a circle, and the edges corresponding to the action of the other automaton being chords in that circle, motivating the title of this section.

There is an easy sufficient condition for μ to be not only Markov, but uniform Bernoulli on a cylinder. To state it, we need to make several definitions:

Definition 4.3.1: $\sigma : X^{\mathbf{N}} \to X^{\mathbf{N}}$ is the **shift**, defined by $\sigma(aw) = w$ for $w \in X^{\mathbf{N}}$.

Definition 4.3.2: when L is a Moore machine, the **delayed automaton** σL is the automaton that computes $\sigma \circ L$. It has the same states, initial state and the transition function as L, and the output function $\sigma\lambda$ given by

$$\sigma\lambda(s, x) = \lambda(\pi(s, x)),$$

which is well-defined when L is a Moore machine.

Remark 4.3.3. *When L is Moore, for any finite word $w \in X^*$ and $x \in X$,*

$$L(wx) = L(0)\sigma L(w) = L(1)\sigma L(w).$$

Proposition 4.3.4. *Let X be a finite alphabet. Let L be a Moore machine with initial state*

s_0, and let $a = \lambda(s_0)$. Let ν be the uniform Bernoulli measure on $X^{\mathbf{N}}$.

Then $\mu = L_*\nu$ is supported on the cylinder $aX^{\mathbf{N}}$, and $\mu|_a = (\sigma L)_*\nu$. If σL is invertible, $\mu|_a$ is uniform Bernoulli (i.e. $\mu|_a = \nu$).

Proof. By 3.2.2, $\mu(aX^{\mathbf{N}}) = 1$ when L is Moore. Now by $\mu_a = (L_*\nu)|_a = (\sigma L)_*\nu$, since for all $v \in X^*$,

$$
\begin{aligned}
(\sigma L)_*\nu(vX^{\mathbf{N}}) &= \nu\left((\sigma L)^{-1}(vX^{\mathbf{N}})\right) \\
&= \nu\left(L^{-1}(\sigma^{-1}\left((vX^{\mathbf{N}})\right)\right) \\
&= \nu\left(L^{-1}\left(\bigsqcup_{x\in X} xvX^{\mathbf{N}}\right)\right) \\
&= \nu\left(L^{-1}(avX^{\mathbf{N}})\right) \\
&= L_*v\left(avX^{\mathbf{N}}\right) \\
&= (L_*v)|_a(vX^{\mathbf{N}}) \qquad \text{(since } L_*\nu(aX^{\mathbf{N}}) = 1\text{).}
\end{aligned}
$$

Thus $(\sigma L)_*\nu = (L_*\nu)|_a$.

If σL is invertible, $(\sigma L)_*\nu = \nu$ by 3.2.3. This completes the proof. □

Corollary 4.3.5. *Let X, A, B and $L_{A,B}$ be as in Prop 4.2.3 (so B is invertible, and $L_{A,B}$ is Moore). Let ν be the uniform Bernoulli measure on X^*.*

Then $\mu = \log_A(B)_*\nu$ is supported on $L(0)X^{\mathbf{N}}$, and $\mu|_{L(0} = \nu$.

Example 4.3.6. *When A is the adding machine, and B is the Bellaterra automaton (of Fig. 4.2), $L(0) = L(1) = 0$. The delayed automaton σL is in Figure 4.3b, and it is invertible (but not minimal: can be reduced to an automaton with 5 states).*

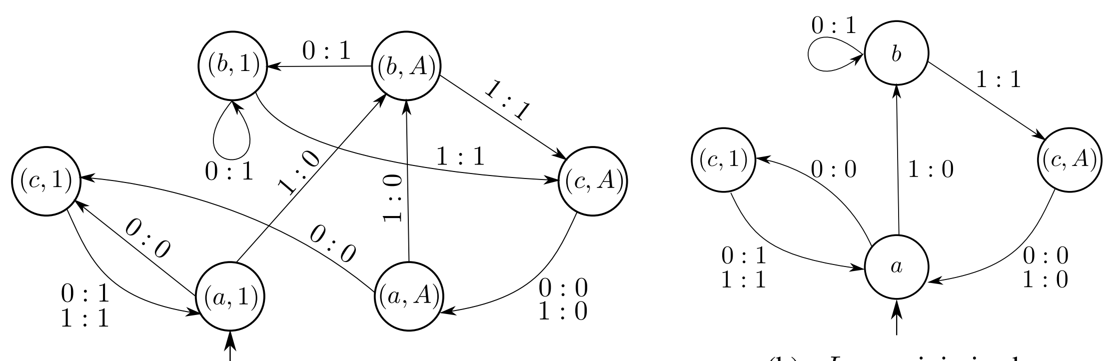
Therefore, $\mu = \log_A(B)_*\nu$ is the uniform Bernoulli measure supported on $0X^{\mathbf{N}}$, i.e. $\mu|_0 = \nu$ and $\mu|_1 = 0$. △

Prop 4.3.4 demonstrates that when B is invertible, the delayed automaton $\sigma L_{A,B}$ can be useful for examining $\mu_{A,B}$. We make use of it again for what follows:

Example 4.3.7. *Let ν be the uniform Bernoulli measure. For A - the adding machine, and B - automaton F (see Figure 2.2b), the measure $\mu = \log_A(B)_*\nu$ is finite-state. Furthremore, $\mu|_0 = 0$, and automaton in Figure 4.5 computes $\mu|_1$.*

Proof. By Prop 4.3.4 and the already computed $L = L_{A,B}$ in Fig 4.1, $\mu|_0 = 0$, and the measure is supported on $1X^{\mathbf{N}}$, with $\mu|_1 = (\sigma L)_*\nu$. We thus point our attention to σL, shown in Figure 4.4a.

First, observe that the automaton σL is not minimal. After identifying states $(a, 1)$ and (a, A) into state a, and identifying states $(b, 1)$ and (b, A) into state b, we obtain a minimal automaton L (Figure 4.4b).



(a) $\sigma L_{A,B}$

(b) $\sigma L_{A,B}$ minimized.

Figure 4.4: Automatons $\sigma L_{A,B}$ and its minimization when B is automaton F

Let us use the names of the states a, b, $(c, 1)$ and (c, A) for the respective actions of the automaton when the corresponding state is chosen as the initial one. Then $\log_A(B) = a$.

If g is an action on the tree $\mathcal{T}$, we write μ_g for $g_*\nu$. Thus we are interested in $\mu_a = \mu = a_*\nu = (\sigma L)_*\nu$, and we compute it by writing down its sections in terms of μ_a, μ_b, $\mu_{c,1}$ and $\mu_{c,A}$.

We apply Corollary 3.2.2 to L to obtain the sections by one character:

$$\mu_a|_0 = \frac{\mu_b + \mu_{c,1}}{2} \qquad \mu_a(0X^{\mathbf{N}}) = 1$$

$$\mu_a|_1 = 0 \qquad \mu_a(1X^{\mathbf{N}}) = 0$$

$$\mu_b|_0 = 0 \qquad \mu_b(0X^{\mathbf{N}}) = 0$$

$$\mu_b|_1 = \frac{\mu_b + \mu_{c,A}}{2} \qquad \mu_b(1X^{\mathbf{N}}) = 1$$

$$\mu_{c,1}|_0 = 0 \qquad \mu_{c,1}(0X^{\mathbf{N}}) = 0$$

$$\mu_{c,1}|_1 = \mu_a \qquad \mu_{c,1}(1X^{\mathbf{N}}) = 1$$

$$\mu_{c,A}|_0 = \mu_a \qquad \mu_{c,A}(0X^{\mathbf{N}}) = 1$$

$$\mu_{c,A}|_1 = 0 \qquad \mu_{c,A}(1X^{\mathbf{N}}) = 0$$

Having expressed the sections by one character in terms of each other, we have obtained a set of recursive relations which allows us to compute sections by arbitrary words. To find the set of all sections, we proceed by repeatedly computing sections using Prop. 2.2.8. We find:

$$\frac{\mu_b + \mu_{c,1}}{2}|_0 = 0 \qquad \frac{\mu_b + \mu_{c,1}}{2}(0X^{\mathbf{N}}) = 0$$

$$\frac{\mu_b + \mu_{c,1}}{2}|_1 = \frac{\mu_b + \mu_{c,A} + 2\mu_a}{4} \qquad \frac{\mu_b + \mu_{c,1}}{2}(1X^{\mathbf{N}}) = 1$$

$$\frac{\mu_b + \mu_{c,A}}{2}|_0 = \mu_a \qquad \frac{\mu_b + \mu_{c,A}}{2}(0X^{\mathbf{N}}) = \frac{1}{2}$$

$$\frac{\mu_b + \mu_{c,A}}{2}|_1 = \frac{\mu_b + \mu_{c,A}}{2} \qquad \frac{\mu_b + \mu_{c,A}}{2}(1X^{\mathbf{N}}) = \frac{1}{2}$$

And again:

$$\frac{\mu_b + \mu_{c,A} + 2\mu_a}{4}|_0 = \frac{\mu_a + \mu_b + \mu_{c,1}}{3} \qquad \frac{\mu_b + \mu_{c,A} + 2\mu_a}{4}(0X^{\mathbf{N}}) = \frac{3}{4}$$
$$\frac{\mu_b + \mu_{c,A} + 2\mu_a}{4}|_1 = \frac{\mu_b + \mu_{c,A}}{2} \qquad \frac{\mu_b + \mu_{c,A} + 2\mu_a}{4}(1X^{\mathbf{N}}) = \frac{1}{4}$$

Finally:

$$\frac{\mu_a + \mu_b + \mu_{c,1}}{3}|_0 = \frac{\mu_b + \mu_{c,1}}{2} \qquad \frac{\mu_a + \mu_b + \mu_{c,1}}{3}(0X^{\mathbf{N}}) = \frac{2}{3}$$
$$\frac{\mu_a + \mu_b + \mu_{c,1}}{3}|_1 = \frac{\mu_b + \mu_{c,A} + 2\mu_a}{4} \qquad \frac{\mu_a + \mu_b + \mu_{c,1}}{3}(1X^{\mathbf{N}}) = \frac{1}{3}$$

Since we have obtained no new sections at this step, the sections so far are all the sections of μ. We have all the data now to build the automaton in Figure 4.5 that computes $\mu|_1$. $\square$

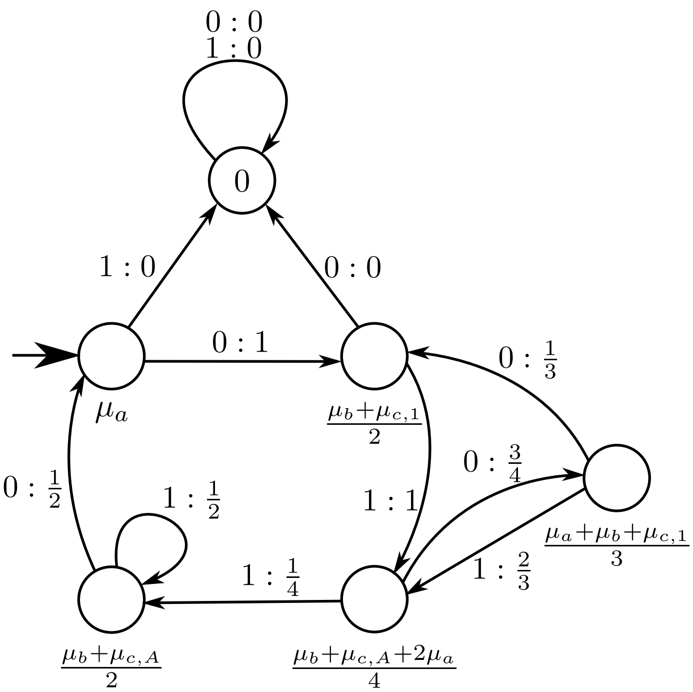


Figure 4.5: Automaton that computes $\mu_{A,B}|_1$ for A - odometer, B - automaton F

It should be noted that $\mu_{A,B}$, while being quasi-finite-state, is not necessarily finite-state:

Example 4.3.8. *Let A be the adding machine, and B be the Lamplighter automaton; see Figure 4.6. We show that the measure $\overline{\mu}$ is not finite-state.*

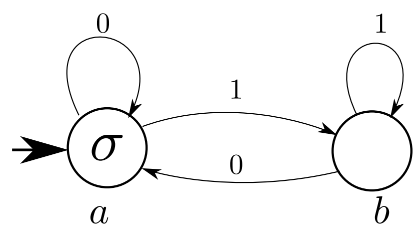


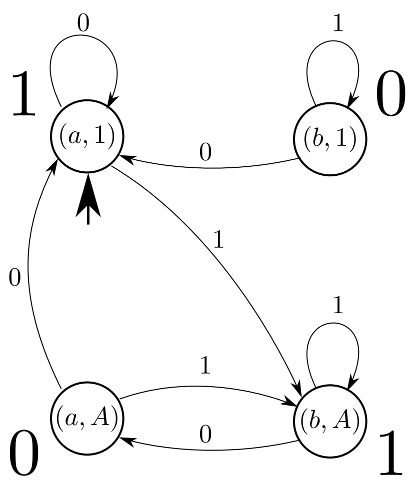
Figure 4.6: The lamplighter automaton

$$\begin{array}{cccc}
 & & \pi((a,1),0) = (a,1) & \pi((b,1),0) = (a,1) \\
\lambda((a,1)) = 1 & \lambda((b,1)) = 0 & \pi((a,1),1) = (b,A) & \pi((b,1),1) = (b,1) \\
\lambda((a,A)) = 0 & \lambda((b,A)) = 1 & \pi((a,A),0) = (a,1) & \pi((b,A),0) = (a,A) \\
 & & \pi((a,A),1) = (b,A) & \pi((b,A),1) = (b,A)
\end{array}$$

The computed automata $L_{A,B}$ and $\sigma L_{A,B}$ are in Figures 4.7a and 4.7b, respectively.

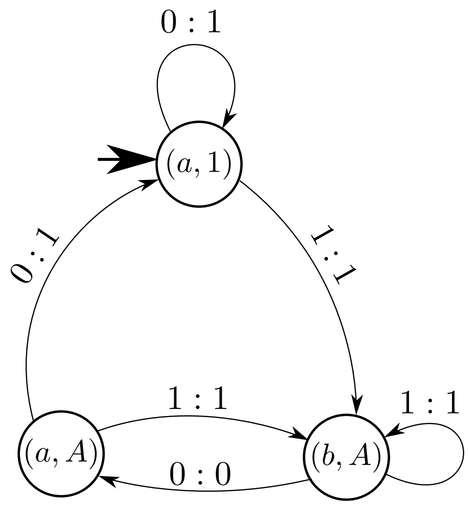
Since $(b,1)$ is not reachable from the initial state $(a,1)$, it is omitted in Figure 4.7b. The automaton in that figure is not minimal; states $(a,1)$ and (a,A) can be identified. The minimized automaton is shown in Figure 4.7c; the relabeling is $a = (a,1) = (a,A)$, $b = (b,A)$, and $(b,1)$ is discarded since as unreachable from the initial state a.

Noting that $\mu_{A,B}$ is supported on $1X^{\mathbf{N}}$ (by Prop. 4.3.4), we now point our attention to the measure $\tilde{\mu} = \mu_{A,B}|_1$. Using Corollary 3.2.2 for the minimized σL in Figure 4.7c and the notation of Example 4.3.7:



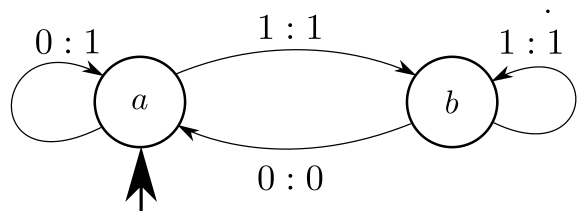
(a) $L_{A,B}$

.



(b) σL; states unreachable from the initial state $(a,1)$ not shown

.



(c) σL minimized

.

Figure 4.7: $L = L_{A,B}$ for A-adding machine, B-lamplighter.

$$\mu_a|_0 = 0 \qquad \mu_a(0X^{\mathbf{N}}) = 0$$

$$\mu_a|_1 = \frac{1}{2}(\mu_a + \mu_b) \qquad \mu_a(1X^{\mathbf{N}}) = 1$$

$$\mu_b|_0 = \mu_a \qquad \mu_b(0X^{\mathbf{N}}) = \frac{1}{2}$$

$$\mu_b|_1 = \mu_b \qquad \mu_b(1X^{\mathbf{N}}) = \frac{1}{2}$$

Now let $\mu_0 := \mu_a$ and $\mu_n := \mu_{n-1}|_1$. Again we use Prop. 2.2.8:

$$\begin{aligned}
\mu_1 &= \frac{(\mu_a + \mu_b)}{2} \\
\mu_2 = \mu_1|_1 &= \frac{1}{2}\left(\mu_a|_1 + \frac{\mu_b|_1}{2}\right) / \mu_1\left(1X^{\mathbf{N}}\right) \\
&= \frac{(\mu_a + 2\mu_b)}{4} \cdot \frac{4}{3} \\
&= \frac{(\mu_a + 2\mu_b)}{3} \\
\mu_3 &= \frac{(\mu_a + 3\mu_b)}{4} \\
&\ldots
\end{aligned}$$

We have the following:

Proposition 4.3.9.

$$\begin{aligned}
\mu_n &= \frac{\mu_a + n\mu_b}{n+1} \\
\mu_n(0X^{\mathbf{N}}) &= \frac{n}{2(n+1)} \\
\mu_n(1X^{\mathbf{N}}) &= \frac{n+2}{2(n+1)}
\end{aligned}$$

Proof. By induction. The proposition holds for $n = 1$. Assuming it holds for $k = n$,

$$\begin{aligned}
\mu_{k+1} = \mu_k|_1 &= \frac{1}{k+1}\left(\frac{\mu_a + \mu_b}{2} + \frac{1}{2}k\mu_b\right) / \left(\frac{k+2}{2(k+1)}\right) \\
&= \frac{\mu_a + (k+1)\mu_b}{k+2}. \ \square
\end{aligned}$$

Note that measures μ_n are all distinct.

Corollary 4.3.10. *$\mu_{A,B}$ is not finite-state when A is the adding machine and B is Lamplighter.*

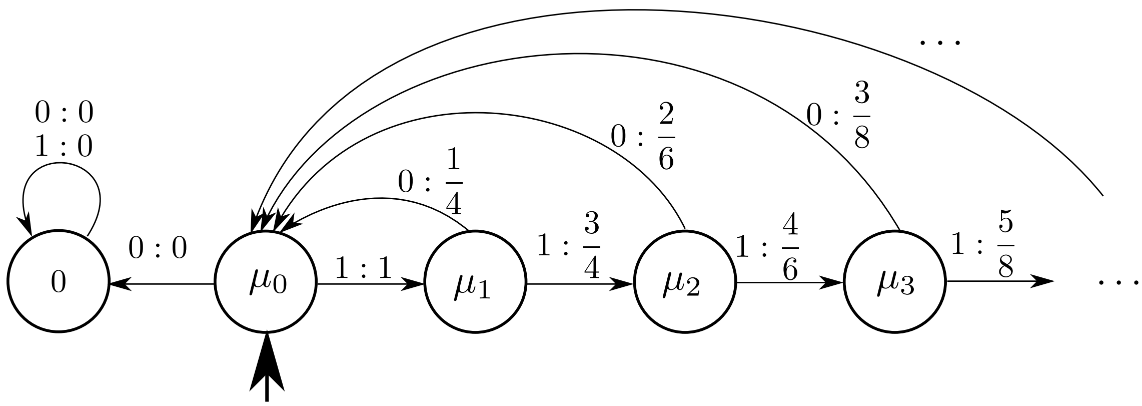


Figure 4.8: The infinite automaton computing $\tilde{\mu}_{A,B}$ where A is the adding machine, and B is the Lamplighter automaton.

Corollary 4.3.11. *μ_n for $n = 0, 1, 2, \ldots$ are all the nontrivial sections of $\tilde{\mu}$.*

Proof. This immediately follows from observing that $\mu_n|_0 = \mu_0$ for $n > 0$:

$$\mu_n|_0 = \frac{\mu_a + n\mu_b}{n+1}|_0 = \frac{1}{n+1}\frac{n\mu_a}{2}\frac{2(n+1)}{n} = \mu_a = \mu_0.$$

The (infinite) automaton that computes $\tilde{\mu}$ is shown in Figure 4.8. $\triangle$

Observe that the computations in these examples are almost linear. To make this notion precise:

Proposition 4.3.12. *Let $X = \{x_0, \ldots, x_{k-1}\}$ be a finite alphabet, L be a Mealy machine with states $S = \{g_0, \ldots, g_{n-1}\}$, $v = (a_0, a_1, \ldots, a_{n-1})$, $a_i \in \mathbf{R}$, and ν - a Bernoilli measure given by vector $p = (p_0, \ldots, p_{k-1})$. Let*

$$\mu_v = \sum_{i=0}^{n-1} a_i g_{i*}\nu.$$

Then for $x \in X$ there exists a matrix $M_x : \mathbb{R}^{n^2} \to [0,1]$ and a vector $p_x : \mathbf{R}^n \to [0,1]$

such that

$$\mu_v|_x = \mu_w$$

with

$$w = \frac{M_x v}{p_x . v}.$$

The entries of M_x and p_x are given by

$$M_x(i,j) = \sum_{y:\pi(g_i,y)=g_j \textit{ and } \lambda(g_i,y)=x} p(y);$$

$$p_x(j) = \sum_{i=0}^{n-1} M_x(i,j).$$

Proof. From 3.2.2 and 2.2.8:

$$\left(\sum_{i=0}^{n} a_i g_{i*}\nu\right)\Bigg|_x = \frac{\sum_{i=0}^{n} a_i g_{i*}\nu(xX^{\mathbf{N}})(g_{i*}\nu)|_x}{\sum_{i=0}^{n} a_i g_{i*}\nu(xX^{\mathbf{N}})}$$

$$= \frac{\sum_{i=0}^{n} a_i \sum_{y\in\lambda_{g_i}^{-1}(x)} p(y)\pi(g_i,y)_*\nu}{\sum_{i=0}^{n} a_i \sum_{y\in\lambda_{g_i}^{-1}(x)} p(y)}.$$

The proposition follows.

Corollary 4.3.13. *Let*

$$\phi_x(v) := \frac{M_x v}{p_x \cdot v}.$$

Then μ_v is finite-state if and only if the orbit of v under the action of $\phi_x : x \in X$ is finite. The graph of action is the transition diagram of the automaton that computes $\mu_{[v]}$.

The above corollary can be made simpler once we consider v as an element of $\mathbf{RP}^n$.

For $v = (a_1, \ldots, a_n)$, write $[v] = [a_0 : a_1 : \ldots a_{n-1}] \in \mathbf{RP}^n$, and write

$$\mu_{[v]} := \frac{\mu_v}{\mu_v\left(X^{\mathbf{N}}\right)}.$$

This is well-defined, and

$$[\phi_x(v)] = [M_x v].$$

Corollary 4.3.14. *$\mu_{[v]}$ is finite-state if and only if the orbit of $[v]$ under the action of the free semigroup generated by $\langle M_x : x \in X\rangle$ is finite.*

In the special case ν - uniform Bernoulli measure, it is convenient to use $\tilde{M}_x(i,j)$ with entries

$$\tilde{M}_x(i,j) = \sum_{y:\pi(g_i,y)=g_j \text{ and } \lambda(g_i,y)=x} 1.$$

Similarly, set $\tilde{p}_x = |X|p_x$. By definition, $\tilde{M}_x = |X|M_x$, $[\tilde{M}_x v] = [M_x v]$, and $\phi_x(v) = \tilde{M}_x v/\tilde{p}_x \cdot v$. However, $\tilde{M}_x$ have integer entries: $M_x(i,j) \in \{0, 1, \ldots, |X|\}$.

Corollary 4.3.15. *When ν- uniform Bernoulli, $\mu_{[v]}$ is finite-state if and only if the orbit of $[v]$ under the action of the free semigroup of integer matrices $\langle \tilde{M}_x : x \in X\rangle$ is finite.*

Example 4.3.16. *When $L = L_{A,B}$, where A is the adding machine, and B is automaton F , we have $\tilde{M}_0$ and $\tilde{M}_1$ as follows:*

$$\tilde{M}_0 = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\tilde{p}_0 = (2, 0, 0, 2)$$

$$\phi_0(v) = \tilde{M}_0 v / \tilde{p}_0 \cdot v$$

$$\tilde{M}_1 = \begin{pmatrix} 0 & 0 & 2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$$p_1 = (0, 2, 2, 0)$$

$$\phi_1(v) = M_1 v / p_1 \cdot v$$

The orbit of $(1, 0, 0, 0)$ *under the action of* $\langle \phi_0, \phi_1 \rangle$ *is*

$$((0,0,0,0),(0,1/2,0,1/2),(0,1/2,1/2,0),(1/3,1/3,1/3,0),(1/2,1/4,0,1/4),(1,0,0,0)).$$

These correspond to the states in Figure 4.5.

Equivalently, the orbit of $[1:0:0:0]$ *under the action of* $\langle \tilde{M}_0, \tilde{M}_1 \rangle$ *is*

$$([0:0:0:0]:[0:1:0:1]:[0:1:1:0]:[1:1:1:0]:[2:1:0:1]:[1:0:0:0]).$$

$\triangle$

Example 4.3.17. *When $L = L_{A,B}$ with A-adding machine, and B - Lamplighter, we have*

$$\tilde{M}_0 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \qquad \tilde{M}_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

The orbit of $[1:0]$ under the action of $\tilde{M}_1$ is $\{[1:n] \ : \ n \in \mathbf{N}\}$, and is not finite. $\triangle$

4.4 When the measure is Markov

Having obtained the automaton that computes a measure, we can ask the question of what kind of measure it is. Here, we specialize Theorem 3.4.1 to the images of Bernoulli measures under the automatic logarithm map.

Theorem 3.4.1 is illustrated by the following:

Example 4.4.1. *When A is the odometer and B is automaton F, measure $\mu_{A,B}$ is the 3-step Markov measure defined in Table 4.3. The admissible words for measure $\mu_{A,B}$ are all words not containing 000 or 1101. For all admissible w, $|mu|_w$ is determined by its suffix of length 3.* $\triangle$

Proof. We know it is k-step Markov for some $k < 21$, since the automaton M in Figure 4.5 satisfies the conditions of Theorem 3.4.1. The result follows from examining M. $\square$

w ends in	$\mu\|_w$
00	$\frac{\mu\|_b + \mu\|_{c,1}}{2}$
11	$\frac{\mu\|_b + \mu\|_{c,A}}{2}$
01	$\frac{\mu\|_b + \mu\|_{c,A} + 2\mu_a}{4}$
110	μ_a
010	$\frac{\mu_a + \mu\|_b + \mu\|_{c,1}}{3}$

Minimal forbidden words: 000, 1101.

Table 4.3: $\mu_{A,B}$ as a Markov measure when B is automaton F

5. IMAGES OF MARKOV MEASURES

In this chapter, we turn to examining images of Markov measures under the action of tree endomorphisms, generalizing an approach (due to Kravchenko) that involves constructing a certain lifting.

This approach is different from the one in Section 3.2, and yields some insights not readily available with the approach of that section, such as absolutely continuity when g is polynomial-activity.

Note that a strongly-connected tree automorphism, in general, is not polynomial-activity: if the graph of the automaton contains two disjoint cycles, the activity is exponential. Therefore, we consider these two cases separately.

From here onwards, we assume the automorphisms and the automata are all finite-state.

5.1 Outline

Let X be a finite alphabet, and $(X^{\mathbf{N}}, \mathcal{B}, \mu)$ a probability measure space such that the shift $\sigma : X^{\mathbf{N}} \to X^{\mathbf{N}}$ is a probability-preserving transformation. If σ is an ergodic probability-preserving transformation on $(X^{\mathbf{N}}, \mathcal{B}, \mu)$, Birkhoff's Pointwise Ergodic Theorem is a tool that can be used to calculate frequencies: for almost all $w \in X^{\mathbf{N}}$ and $x \in X$,

$$\mathrm{freq}(x) = \lim_{n\to\infty} \sum_{k=0}^{n-1} \chi_x \circ \sigma^k(w) = \int \chi_x d\mu,$$

where χ_x is the characteristic function on the cylinder $xX^{\mathbf{N}}$, and the left hand side of the equation is the frequency of x.

Example 5.1.1. *let $X = \{0, 1\}$, and μ be a Bernoulli measure defined by the probability vector $q = (p, 1-p)$. On the cylinders, μ is given as follows: $\mu(0wX^{\mathbf{N}}) = p\mu(wX^{\mathbf{N}})$*

and $\mu(1wX^{\mathbf{N}}) = (1-p)\mu(wX^{\mathbf{N}})$ *for any finite word* w*, and* $\mu(X^{\mathbf{N}}) = 1$. $\triangle$

This measure can be understood as a (stochastic) process of flipping a coin, possibly biased, which gives heads with probability p. One would expect to get $\mathrm{freq}(0) = p$; and indeed, by the ergodic theorem we have almost everywhere:

$$\mathrm{freq}(0) = \int \chi_0 d\mu = \mu(0X^{\mathbf{N}}) = p\mu(X^{\mathbf{N}}) = p.$$

We then may ask the question: how do actions on the space affect frequencies? A natural object that acts on infinite sequences is a Mealy machine: an initial finite-state automaton with output, so we restrict our attention to the action $w \mapsto gw$ given by an automaton transformation, and examine

$$\lim_{n\to\infty} \sum_{k=0}^{n-1} \chi_x \circ \sigma^k(gw).$$

The case for μ - a Bernoulli process (on a two-letter alphabet) was solved by Ryabinin in [2], and for μ - a Bernoulli measure on a finite alphabet was answered by Kravchenko in [3]. We generalize these result to the case of μ - a Markov measure.

5.2 Automorphisms of polynomial activity

In the polynomial activity case, the pushforward of any non-atomic measure (measures ν such that $\nu(S) = 0$ for any countable set S) is easy to find. This comes from the following lemmas:

Lemma 5.2.1. *let* $A = (X, S, \pi, \lambda)$ *be a strongly connected automaton, and* $g(w) := \lambda(g, w)$ *as above. A* ***nontrivial simple cycle*** *through* $s \in S$ *is a path in* A *that starts and ends in* s *and consists of nontrivial states. Then* g *has polynomial activity only if there do not exist two distinct nontrivial cycles through any state* $s \in S$.

Proof. suppose that there exists a state s with two distinct cycles, defined by input words a and b, through it:

$$s \xrightarrow[\lambda(s,a_1)]{a_1} s_1 \xrightarrow[y_2]{a_2} \dots \xrightarrow[\lambda(s_n,a_n)]{a_n} s_{n+1} = s;$$
$$s \xrightarrow[\lambda(s,b_1)]{b_1} t_1 \xrightarrow[y_2]{b_2} \dots \xrightarrow[\lambda(s_m,b_m)]{b_m} t_{m+1} = s.$$

Furthermore, note that since A is strongly connected, there exists a path defined by an input word w_{gs} that goes from g to s.

Consider words w of the form

$$w = w_{gs} A_1 A_2 \dots A_{2n},$$

where $A_i \in \{a, b\}$, and where a occurs n times (i.e. $|\{i \ : \ A_i = a\}| = n$). All these words have the same length:

$$|w_{gs} A_1 A_2 \dots A_{2n}| = |w_{gs}| + n(|a| + |b|),$$

and this length is linear in n.

However, there are at least 2^n distinct such words. Indeed, there are 2^n distinct words in the alphabet $\{a, b\}$; it suffices to show that they give distinct words in X.

Since input words a and b define distinct paths in A, there exist i, j such that $s_i \neq t_j$.

Given an input word w, define a function $F : X^{\mathbf{N}} \to \{0,1\}^{\mathbf{N}}$ as follows:

$$F(wx) = \begin{cases} F(w)0, \text{ if } \pi(s, wx) = s_i; \\ F(w)1, \text{ if } \pi(s, wx) = t_j; \\ F(w)\text{otherwise.} \end{cases}$$

That is, F simply writes down 0 when s_i is encountered, and 1 when t_j is encountered while going along the path defined by w.

Let R be the rewriting of a word in alphabet $\{a, b\}$ in X. Then $F \circ R$ is injective by the assumption that s_i and t_j are distinct states such that s_i is not on the path defined by b, and t_j is not on the path defined by a. Therefore, R must be injective as well.

Finally, we note that restriction of g to w is nontrivial, since $\pi(g, w) = s$, a nontrivial state.

We thus produced, for arbitrary n, at least 2^n words w of length $|w_{gs} + n(|a| + |b|)$ such that $g|_w$ is nontrivial; by definition, this implies exponential activity. The lemma holds by contradiction. □.

Lemma 5.2.2. A^{-1} *has polynomial activity whenever* A *has polynomial activity.*

Proof. this is by definition from the following observation:

$$g|_w \neq 1 \Leftrightarrow g^{-1}|_{g(w)} \neq 1.$$

(In other words, if g leaves suffixes of w unchanged, so does g^{-1}).

Lemma 5.2.3. *(Kravchenko) Let V, V_{max} be given by*

$$V = \{w \in X^{\mathbf{N}} \;:\; g^{-1}|_w = \mathbf{1}\}$$

$$V_{max} = \{w \in V \;:\; w = vw', |w'| > 0 \Rightarrow v \notin V\}.$$

Then μ is supported on $\sqcup_{v \in V_{max}} vX^{\mathbf{N}}$.

That is, V is the set of words giving trivial sections, and words in V_{max} are words in V whose proper prefixes are not in V (i.e. yield non trivial sections). In a tree diagram of the automorphism g^{-1}, $v \in V_{max}$ if next-to-last node on the path given by v is a switch node, and there are no switches in its subtree.

Proof. First, note that the cylinders $vX^{\mathbf{N}}$ are disjoint for $v \in V_{max}$: if $v_1X^{\mathbf{N}}$ and $v_2X^{\mathbf{N}}$ intersect, than either v_1 starts with v_2, or v_2 starts with v_1 - neither is possible by construction. Therefore, the union of sets $vX^{\mathbf{N}}$ for $v \in V_{max}$ is disjoint.

Now, to prove the lemma, we show that

$$W = X^{\mathbf{N}} - \cup_{v \in V_{max}} vX^{\mathbf{N}}$$

is at most countable. Indeed, let $w \in W$. Then the path in the statues of that automaton of g^{-1} defined by w must consist of nontrivial states: if $\pi(g, w_1w_2w_3 \ldots w_n)$ is a trivial state, then so are all the subsequent states on the path defined by w, and so $g^{-1}|_{w_1w_2\ldots w_n}$ is trivial.

By assumption, A (and thus A^{-1}) is finite, and so the path in A^{-1} defined by w must pass through some non-trivial state s infinitely often. By Lemma 5.2.2, A^{-1} has polynomial activity. There, Lemma 5.2.1 applies, and so there is at most one nontrivial cycle passing through s. Therefore w is eventually periodic. The set of such words is countable. Since for Markov measures, $\mu(S) = 0$ when S is countable, the lemma follows. $\square$

To proceed further, we introduce a technical definition. Given a measure μ and an automorphism g, we say g is **μ-V_{max}-compatible** if $\mu(vX^{\mathbf{N}}) \neq 0$ for all $v \in V_{max}$.

If μ is a Markov measure induced by a matrix L, this means that all paths in the automaton of restrictions g that lead to a trivial state must be induced by words w that are not forbidden in L (i.e. $L_{w_i w_{i+1}} > 0$ for $i = 0..|w| - 1$).

We can now show the following:

Theorem 5.2.4. *Let μ be a Markov measure induced by $L,\ l$, and g - an automorphism of polynomial activity. If g is μ-V_{max}-compatible, then $g_*\mu$ is absolutely continuous w.r.t. μ, with the derivative given by*

$$\frac{dg_*\mu}{d\mu} = \sum_{v \in V_{max}, a \in X} \frac{\mu(g^{-1}(vX^\infty)}{\mu(vX^\infty)} \frac{L(g^{-1}(v)_{|v|}, a)}{L(v_{|v|}, a)} \chi_{vaX^{\mathbf{N}}}.$$

Proof. note that the above expression is well-defined iff g is μ-V_{max}-compatible.

Now, extending the approach of Kravchenko to Markov measures, let

$$g' = \sum_{v \in V_{\max}, a \in X} \frac{\mu(g^{-1}(vX^\infty))}{\mu(vX^\infty)} \frac{L(g^{-1}(v)_{|v|}, a)}{L(v_{|v|}, a)} \chi_{vaX^{\mathbf{N}}}.$$

By construction, the measure $g'd\mu$ is supported on cylinder sets $vX^{\mathbf{N}}$, for $v \in V_{\max}$. From Lemma 5.2.3, it suffices to show that $dg_*\mu$ and $g'd\mu$ agree on these cylinder sets for the theorem to hold.

Since both measures are continuous, the above will follow if

$$\int_{wX^{\mathbf{N}}} g' d\mu = \int_{wX^{\mathbf{N}}} dg_*\mu$$

for all $w \in V$.

Now, if $w \in V$, then either $w \in V_{max}$, or $w = vaw'$ where $v \in V_{max}$ and $a \in X$. In the latter case:

$$
\begin{aligned}
\int_{wx^{\mathbf{N}}} g' d\mu &= \frac{\mu(g^{-1}(vX^{\mathbf{N}}))}{\mu(vX^{\mathbf{N}})} \frac{L(g^{-1}(v)_{|v|}, a)}{L(v_{|v|}, a)} \mu(vaw'X^{\mathbf{N}}) \\
&= \frac{\mu(g^{-1}(vX^{\mathbf{N}}))}{\mu(vX^{\mathbf{N}})} \frac{L(g^{-1}(v)_{|v|}, a)}{L(v_{|v|}, a)} \\
&\cdot l(v_1)L(v_1, v_2) \dots L(v_{|v|-1}, v_{|v|}) \cdot L(v_{|v|}, a) \cdot L(a, w'_1) \dots L(w'_{|w'|-1}, w'_{|w'|}) \\
&= \frac{\mu(g^{-1}(vX^{\mathbf{N}}))}{\mu(vX^{\mathbf{N}})} \frac{L(g^{-1}(v)_{|v|}, a)}{L(v_{|v|}, a)} \cdot \mu(vX^{\mathbf{N}}) \cdot L(v_{|v|}, a) \cdot L(a, w'_1) \dots L(w'_{|w'|-1}, w'_{|w'|}) \\
&= \mu(g^{-1}(vX^{\mathbf{N}}))L(g^{-1}(v)_{|v|}, a) \cdot L(a, w'_1) \dots L(w'_{|w'|-1}, w'_{|w'|}) \\
&= \mu(g^{-1}(v)aw'X^{\mathbf{N}}) = \mu(g^{-1}(vaw'X^{\mathbf{N}})) \qquad (5.1) \\
&= \mu(g^{-1}(wX^{\mathbf{N}})) \\
&= \int_{wx^{\mathbf{N}}} g_*\mu.
\end{aligned}
$$

In the above, equality 5.1 follows from the assumption that $v \in V_{\max}$, and therefore

$$g^{-1}(vaw') = g^{-1}(v)aw'$$

for all w'.

Now that we have verified that $g'd\mu$ and $dg_*\mu$ agree on cylinders $wX^{\mathbf{N}}$ for $w \in V$, the theorem holds.□

5.3 Subexponential case

It is known that finite-state automorphisms have either polynomial, or exponential activity.

However, the result of the Theorem 5.2.4 also holds in the case of $g \in \operatorname{Aut} \mathcal{T}$, g – **not** finite-state, having **subexponential activity**, i.e. g satisfying the following condition:

$$|\{w\ :\ |w| = n,\ g|_w \neq 1\}| < C^n \text{ for all } C > 1.$$

Indeed, a more general version of Lemma 5.2.3 holds:

Lemma 5.3.1. *The conclusion of Lemma 5.2.3 holds in the subexponential case.*

Proof. all we need to demonstrate is that $\mu(X^{\mathbf{N}} - \sqcup_{v \in V_{max}} vX^{\mathbf{N}}) = 0$. Let $V_n = \{w\ :\ |w| = n, g|_w \neq 1\}$. Following [Grigorchuk-Dudko], let

$$M_n = \mu(\sqcup_{w \in V_n} wX^{\mathbf{N}}),$$

and let $L_{max} = \max L_{ij} < 1$. Then

$$\begin{aligned} M_n &\leq |V_n| \max\{\mu(wX^{\mathbf{N}})\ :\ w \in V_n\} \\ &< |V_n| L_{max}^n. \end{aligned}$$

For $w \in X^{\mathbf{N}}$, $w \neq vw'$ for some $v \in V_{max}$ if and only if $w_1 w_2 \dots w_n \in V_n$ for $n = 1..|w|$, and so

$$\mu(X^{\mathbf{N}} - \sqcup_{v \in V_{max}} vX^{\mathbf{N}}) = \lim_{n \to \infty} M_n = 0$$

by the definition of subexponential activity of an automaton.□

Aside from Lemma 5.2.3, the proof of Theorem 5.2.4 does not depend on whether g has polynomial activity or not; all that matters is that the conclusion of Lemma 5.2.3 holds. Therefore, we have:

Corollary 5.3.2. *The conclusion of Theorem 5.2.4 holds in the subexponential case.*

Remark 5.3.3. *Finite automata with sub-exponential activity always have polynomial ac-*

tivity. The result above applies to automorphisms which are not given by finite automata.

5.4 Strongly-connected automata

The measure given by μ is invariant and ergodic. Naively, one could hope that the automaton action preserves this, and Birkhoff's Pointwise Ergodic Theorem could be applied directly by a change of variable in the integral:

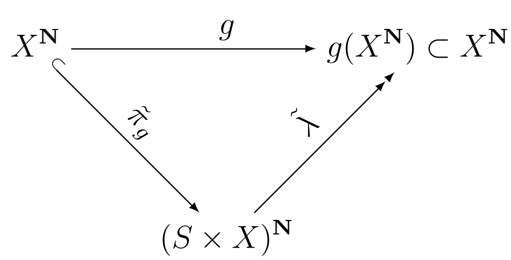
$$\mathrm{freq}(x) = \lim_{n\to\infty} \sum_{k=0}^{n-1} \chi_x \circ \sigma^k(gw) \overset{?}{=} \int \chi_x dg_*\mu.$$

However, the pushforward measure $g_*\mu$ is not a-priori invariant and ergodic, and the measure is not easy to deal with directly. On the cylinders, it is given by

$$g_*\mu(y_1y_2\dots y_nX^{\mathbf{N}}) = \sum_{g\xrightarrow[y_1]{x_1}s_1\xrightarrow[y_2]{x_2}\dots\xrightarrow[y_n]{x_n}s_n} l(g)L(g,x_1)L(x_1,x_2)\dots L(x_{n-1},x_n),$$

where the summation is over *all paths* $g \xrightarrow[y_1]{x_1} s_1 \xrightarrow[y_2]{x_2} \dots \xrightarrow[y_n]{x_n} s_n$ in the automaton.

To address this difficulty, we keep track of the states A goes through along with the output. We will define maps to obtain a commutative diagram:



with $g_*\mu = \tilde{\lambda}_*\tilde{\pi}_{g*}\mu$. We then define an invariant, ergodic measure Q on $X^{\mathbf{N}}$ which is a pushforward of a Markov measure under the 1-block factor map $\tilde{\lambda}$ such that $g_*\mu \ll Q$. We are then able to apply the pointwise ergodic theorem with respect to measure Q, and the result will hold for μ-almos-all input words w.

Let $\tilde{\pi} : S \times X^{\mathbf{N}} \to (S \times X)^{\mathbf{N}}$ be given by

$$\tilde{\pi}(g, xw) = (g, x)\tilde{\pi}(\pi(g, x), w),$$

and write $\tilde{\pi}_g(w) := \tilde{\pi}(g, w)$.

Given a sequence $\tilde{\pi}(g, w) \in (S \times X)^{\infty}$, we can extract the output $g(w) = \lambda(g, w)$ simply by looking at the states; so we define $\tilde{\lambda} : (S \times X)^{\infty} \to X^{\infty}$ recursively by

$$\tilde{\lambda}((s, x)w) = \lambda(s, x)\tilde{\lambda}(w)$$

for $w \in (S \times X)^{\infty}$.

That is, for each path

$$g \xrightarrow[y_1]{x_1} s_1 \xrightarrow[y_2]{x_2} \dots \xrightarrow[y_n]{x_n} s_n$$

we have

$$\tilde{\pi}_g(x_1 x_2 \dots x_n) = (g, x_1)(s_1, x_2) \dots (s_{n-1}, x_n);$$
$$\tilde{\lambda}(s_0, x_1)(s_1, x_2) \dots (s_{n-1}, x_n) = y_1 y_2 \dots y_n.$$

By construction, we have $\tilde{\lambda} \circ \tilde{\pi}_g(w) = g(w)$, and thus have the commutative diagram from the previous section.

The measure $\tilde{\pi}_{g*}\mu$ on $(S \times X)^{\mathbf{N}})$ is easier to work with then the pushforward $g_*\mu$ on

$X^{\mathbf{N}}$. On the cylinders, it is given as follows:

$$\tilde{\pi}_{g*}\mu((g,x_1)(s_1,x_2)\dots(s_{n-1},x_n)(S\times X)^\infty) =$$
$$= \begin{cases} l(x_1)L_{x_1x_2}\dots L_{x_{n-1}x_n}, \text{if } g \xrightarrow[y_1]{x_1} s_1 \xrightarrow[y_2]{x_2} \dots \xrightarrow[y_n]{x_n} s_n \text{ is a path ;} \\ 0, \text{otherwise.} \end{cases}$$

This is *almost* a Markov measure! The product matches a Markov measure except for the first term, $l(x_i)$. It is piecewise-Markov, scaled by constants on cylinders $(s,x)(S\times X)^\infty$).

To make this statement more precise, let $T = T_{L,A}$ be a transition matrix with entries indexed by elements of $S \times X$, and values given by

$$T_{(s_0,x_0)(s_1,x_1)} = \begin{cases} L(x_0,x_1), \text{if } \pi(s_0,x_0) = s_1; \\ 0, \text{otherwise.} \end{cases}$$

T then is an stochastic matrix:

$$\sum_{(t,y)} T_{(s,x)(t,y)} = \sum_y L(x,y) = 1,$$

since for $T_{(s,x)(t,y)} \neq 0$, t must be uniquely given by $t = \pi(s,x)$.

Example 5.4.1. *Let $X = \{0,1,2\}$, μ be induced by the matrix*

$$L = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 0 & 1/2 & 1/2 \\ 1/2 & 0 & 1/2 \end{pmatrix},$$

and let A have the transition function given by the diagram in Figure 5.1.

Note that 02 is a forbidden word in the subshift defined by L (that is, $\mu(02X^{\mathbf{N}}) =$

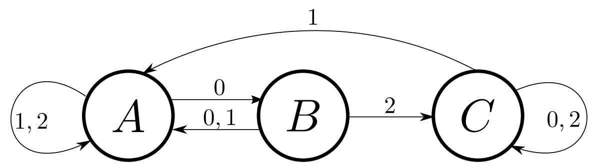


Figure 5.1: Any path from A to C ends in 02

0*).However, the only arrow going into* B *is labeled by* 0*. Therefore,* T *has a zero column at* $(C, 2)$*, and so the Markov measure induced by it is not irreducible.* $\triangle$

The example above calls for a restriction on A that would make T irreducible.

Definition 5.4.2: we call an automaton A L**-strongly-connected**, if for every pair of states $s, t \in S$ and every pair of symbols $x, y \in X$, there exists word $w \in X^*$ such that $\pi(s, xw) = t$, and xwy is not a forbidden word in the subshift defined by the nonzero entries of L (that is, if $w = w_1 \ldots w_n$, then L_{x,w_1}, $L_{w_n,y}$, and $L_{w_i,w_{i+1}}$ for $i = 1..n-1$ are all nonzero).

Lemma 5.4.3. *The Markov chain defined by* T *is irreducible if and only if the atutomaton* A *is* L*-strongly connected.*

Proof. this follows directly from the definitions given above.

Let t be the stationary probability vector of T, so $tT = t$, which exists since T is irreducible; and let P be the associated Markov measure. Then on the cylinders, P is given by

$$P((s_0, x_0) \ldots (s_n, x_n)(S \times X)^\infty) = \begin{cases} t((s_0, x_0))T_{(s_0,x_0)(s_0,x_1)} \ldots T_{(s_{n-1},x_{n-1})(s_n,x_n)} \\ = t((s_0, x_0))L_{x_0 x_1} \ldots L_{x_{n-1}x_n}, \\ \text{if } S_0 \xrightarrow[y_0]{x_0} \ldots \xrightarrow[y_n]{x_n} s_{n+1} \text{ is a path ;} \\ 0, \text{otherwise.} \end{cases}$$

The measure P is ergodic, since T is irreducible. Furthermore, its definition starts to look like the value of $\tilde{\pi}_{g*}\mu$ on the cylinders. Now we can make precise the statement about $\tilde{\pi}_{g*}\mu$ being piecewise-Markov. Note that the vectors l and t are positive, as they are stationary probability vectors of ergodic Markov chains [7]. Therefore, on the cylinders $\Omega_{g,x} = (g,x)(S \times X)^{\mathbf{N}}$, we have:

$$P((g,x)w(S \times X)^{\mathbf{N}}) = \frac{t(g,x)}{l(x)}\tilde{\pi}_{g*}\mu(w(S \times X)^{\mathbf{N}}),$$

that is,

$$\tilde{\pi}_{g*}\mu|_{\Omega_{g,x}} = \frac{l(x)}{t(g,x)}P|_{\Omega_{g,x}}$$

and since $\tilde{\pi}_{g*}\mu$ is supported on $\Omega_{g,x}$,

$$\tilde{\pi}_{g*}\mu = \sum_x \frac{l(x)}{t(g,x)}P|_{\Omega_{g,x}}.$$

From this observation we have the following:

Lemma 5.4.4. $\tilde{\pi}_{g*}\mu \ll P$.

The measure we are interested in, $g_*\mu$, is given as a pushforward: $g_*\mu = \tilde{\lambda}_*\tilde{\pi}_{g*}\mu$.

We now set $Q := \tilde{\lambda}_*P$. After observing that Q is invariant, ergodic, and $g_*\mu \ll Q$, we are able to apply the Birkhoff Pointwise Ergodic theorem with the measure Q to compute the frequencies, and state that the result holds almost everywhere w.r.t measure μ.

Lemma 5.4.5. *Q is an invariant, ergodic measure whenever P is.*

Proof. these properties are carried over by projections which commute with the shift. Note that the following diagram commutes:

$$
\begin{array}{ccc}
(S \times X)^{\mathbf{N}} & \xrightarrow{\sigma} & (S \times X)^{\mathbf{N}} \\
\Big\downarrow \tilde{\lambda} & & \Big\downarrow \tilde{\lambda} \\
X^{\mathbf{N}} & \xrightarrow{\quad \sigma \quad} & X^{\mathbf{N}}
\end{array}
$$

Therefore,

$$
\begin{aligned}
Q(\sigma^{-1}S) &= P(\tilde{\lambda}^{-1}(\sigma^{-1}S)) \\
&= P(\sigma^{-1}(\tilde{\lambda}^{-1}(S))) \\
&= P(\tilde{\lambda}^{-1}(S) \text{ since } P \text{ is invariant by assumption} \\
&= Q(S),
\end{aligned}
$$

so Q is invariant.

For ergodicity, we need to show that whenever $\sigma^{-1}(E) = E$, $Q(E) = 0$ or 1. If E is shift-invariant ($\sigma^{-1}(E) = E$), then so is $\tilde{\lambda}^{-1}(E)$:

$$\tilde{\lambda}^{-1}(E) = \tilde{\lambda}^{-1}(\sigma^{-1}(E)) = \sigma^{-1}(\tilde{\lambda}^{-1}(E)),$$

where the last equality follows from the commutative diagram above. Now

$$Q(E) = P(\tilde{\lambda}^{-1}(E)) = 0 \text{ or } 1,$$

since P is ergodic, and $\tilde{\lambda}^{-1}(E)$ is shift-invariant. □

Lemma 5.4.6. $g_*\mu \ll Q$

Proof. since $g_*\mu = \tilde{\lambda}_*\tilde{\pi}_{g*}\mu$, and $Q = \tilde{\lambda}_*P$, it suffices to verify that $\tilde{\pi}_{g*}\mu \ll P$. But this was shown in Lemma 5.4.4. □

Finally we have the following:

Theorem 5.4.7. *Let L be a stochastic matrix defining an irreducible Markov measure μ, A - an L-strongly-connected automaton with initial state g, $T_{L,A}$ - the stochastic matrix defined above. Then for all $x \in X$ and almost all $w \in X^{\mathbf{N}}$:*

$$\lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \chi_x \sigma^n(gw) = \sum_{s_0 \xrightarrow[x]{y_0} s_1} t(s_0, y_0),$$

where t is the stationary distribution vector of $T_{L,A}$.

Proof. we have shown the measure Q to be invariant and ergodic. Following Kravchenko: by Pointwise Birkhoff Ergodic theorem, for Q-almost all $v \in X^{\mathbf{N}}$,

$$\begin{aligned} \lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \chi_x \sigma^n(v) &= \int_{X^{\mathbf{N}}} \chi_x \, dQ \\ &= Q(xX^{\mathbf{N}}) \\ &= (\tilde{\lambda}_* P)(xX^{\mathbf{N}}) \\ &= \sum_{s_0 \xrightarrow[x]{y_0} s_1} t(s_0, y_0). \end{aligned}$$

Let $V \subset X^{\mathbf{N}}$ be the set of sequences for which the above does not hold. Since $g_*\mu \ll Q$, $g_*\mu(V) = 0$ as well. Therefore, for $g_*\mu$-almost all $v \in X^{\mathbf{N}}$:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \chi_x \sigma^n(v) = \sum_{s_0 \xrightarrow[x]{y_0} s_1} t(s_0, y_0).$$

Now we let $v = g(w)$. The above equation becomes

$$\lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \chi_x \sigma^n(g(w)) = \sum_{s_0 \xrightarrow[x]{y_0} s_1} t(s_0, y_0), \tag{5.2}$$

and it holds for μ-almost all w (if W is the set of $w \in X^{\mathbf{N}}$ for which 5.2 does **not** hold, then $W = g^{-1}(V)$, and $\mu(W) = \mu(g^{-1}(V)) = g_*\mu(V) = 0$). □

The above theorem immediately generalizes to frequencies of words w. Indeed, replacing $x \in X$ with $u = u_0 u_1 \ldots u_{k-1} \in X^k$, we obtain:

Corollary 5.4.8. *Let μ, g, T, etc. be as in Theorem 5.4.7. Then*

$$\lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \chi_u \sigma^n(gw) = \sum_{s_0 \xrightarrow[w_0]{x_0} s_1 \ldots \xrightarrow[w_{k-1}]{x_{k-1}} s_k} t(s_0, x_0) \prod_{j=1}^{k-1} T_{(s_{j-1}, x_{j-1})(s_j, x_j)}$$

for μ-almost-all $w \in X^{\mathbf{N}}$ (where $\chi_u = \chi_{(uX^{\mathbf{N}})}$).

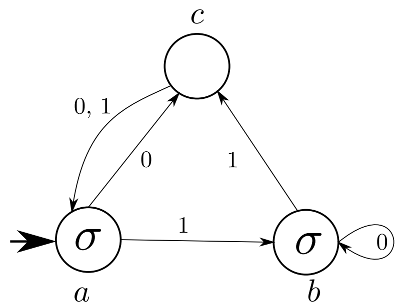
Proof. as in the proof of Theorem 5.4.7,

$$\begin{aligned}
\lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \chi_x \sigma^n(v) &= \int_{X^{\mathbf{N}}} \chi_x \, dQ \\
&= Q(xX^{\mathbf{N}}) \\
&= (\tilde{\lambda}_* P)(uX^{\mathbf{N}}) \\
&= \sum_{s_0 \xrightarrow[w_0]{x_0} s_1 \ldots \xrightarrow[w_{k-1}]{x_{k-1}} s_k} t(s_0, x_0) T_{(s_0, x_0)(s_1, x_1)} \ldots T_{(s_{k-2}, x_{k-2})(s_{k-1}, x_{k-1})}.
\end{aligned}$$

The rest of the proof applies without change. □

Example 5.4.9. *Let $X = \{0, 1\}$, **A** be automaton F with initial state a (already considered* *in Section 3.2), and let* L *define a Bernoulli measure with probabilities* $(1/3, 2/3)$*:*

$$L = \begin{pmatrix} 1/3 & 2/3 \\ 1/3 & 2/3 \end{pmatrix} \quad \boldsymbol{A} =$$



Then T*,* t *and* l *are as follows:*

$$T = \begin{pmatrix} 0 & 0 & \frac{1}{3} & \frac{2}{3} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{2}{3} \\ 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{2}{3} \\ 0 & 0 & \frac{1}{3} & \frac{2}{3} & 0 & 0 \\ \frac{1}{3} & \frac{2}{3} & 0 & 0 & 0 & 0 \\ \frac{1}{3} & \frac{2}{3} & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$t = \begin{pmatrix} \frac{1}{9} & \frac{2}{9} & \frac{1}{9} & \frac{2}{9} & \frac{1}{9} & \frac{2}{9} \end{pmatrix}.$$

$$l = \begin{pmatrix} \frac{1}{3} & \frac{2}{3} \end{pmatrix}.$$

and the frequencies of 0 *and* 1 *in the output are* $5/9$ *and* $4/9$*, respectively.* $\triangle$

Kravchenko observed in [3] that in the case of Bernoulli measures, the vector t can be written as $t = r \otimes l$, where l is the 1-dimensional distribution of the measure, and r is the stationary probality vector of a chain defined by a matrix $S \times S \to \mathbf{R}$ which depends on A and l. In the example above,

$$t = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{3} & \frac{2}{3} \end{pmatrix}$$

With this automaton, this will happen for all matrices L generating a Markov measure.

Example 5.4.10. *When A is automaton F , as above, and L is a stochastic matrix, so*

$$L = \begin{pmatrix} p & 1-p \\ q & 1-q \end{pmatrix},$$

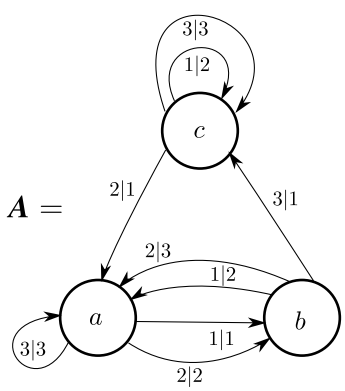
the vector $t' = (q, 1-p, q, 1-p, q, 1-p)$ is an eigenvector of T with eigenvalue 1, so

$$t = \frac{1}{3(1-p+q)}(1,1,1) \otimes (q, 1-p).$$

In general, this will not be the case.

Example 5.4.11. *Let $X = \{1, 2, 3\}$, and L, A (with initial state a) be given as follows:*

$$L = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix} \quad \boldsymbol{A} =$$



Then A is L-strongly-connected, and we have

$$T = \begin{pmatrix} 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix}$$

$$t = \begin{pmatrix} \frac{2}{15} & \frac{2}{15} & \frac{1}{5} & \frac{1}{15} & \frac{2}{15} & \frac{1}{15} & \frac{2}{15} & \frac{1}{15} & \frac{1}{15} \end{pmatrix}$$
$$l = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix}.$$

The frequencies of 0, 1 *and* 2 *in* aw *for* $w \in X^{\mathbf{N}}$ *are* $(4/15, 1/3, 2/5)$*, respectively, for* μ*-almost all* w*.*

In this case, $t \neq v \otimes l$ *for any vector* v*.*

Note that if we modified A*, for example, by making* $\pi(a, 2) = a$*, then* A *would no longer be* L*-strongly-connected: since* $L_{1,3} = \mu(13X^{\mathbf{N}}) = 0$*, this modification disconnects* $(a, 1)$ *and* $(b, 3)$ *in the graph defined by nonzero entries of* T*. In this case,* t *is not positive:*

$$t = \begin{pmatrix} \frac{2}{9} & \frac{2}{9} & \frac{1}{3} & \frac{1}{9} & \frac{1}{9} & 0 & 0 & 0 & 0 \end{pmatrix}.$$

$\triangle$

This example shows that in the case of Markov measures, t cannot depend only on

A and l, like in the case of Bernoulli measures. For a Bernoulli measure with the same 1-dimensional distribution, t would be be positive.

5.4.1 Singularity

In the case of subexponential activity, $g_*\mu$ is absolutely continuous with respect to μ. For tree automorphisms generated by strongly-connected automata, this is not necessarily the case. In [3], Kravchenko considers a sufficient condition for $g_*\mu$ and μ to be singular when μ is a Markov measure. We reproduce his argument, fixing a technical error.

First, let us define K to be a matrix $S \times S \to \mathbf{R}$ with entries given by

$$K_{s,s'} := \sum_{\pi(s,x)=s'} l(x),$$

and let k be is its stationary probability vector ($kK = k, \sum k_i = 1$).

Theorem 5.4.12. *Suppose μ is a Bernouli measure with probability vector p, and g is a strongly-connected tree automorphism. Supppose that there exist $x, y \in X$ such that $p(y) = \max p$, $p(x) < p(y)$, and for some state s, $\lambda(s, x) = y$. Then μ and $g_*\mu$ are singular.*

Proof. in the case of Bernoulli measures, the vector t falls apart as a tensor product:

$$t = k \otimes p : t(s, x) = k(s)p(x)$$

where p is the probability vector of the Bernoulli measure, and k is the stationary probability vector of the Markov chain generated by K defined above (see [3], Lemmas 4 and 5).

Let x, y, s be as above. Then the frequency of y in under $g_*\mu$ is

$$\begin{aligned}\lim_{n\to\infty}\frac{1}{n}\sum_{k=1}^{n}\chi_y\sigma^n(gw) = \sum_{s\xrightarrow[x']{y}s'} t(s,x') &= \sum_{s\xrightarrow[x']{y}s'} q(s)p(x') \\ &< p(y)\sum_{s\xrightarrow[x']{y}s'} q(s) \\ &= p(y)\sum_{s\in S} q(s) \\ &= p(y),\end{aligned}$$

which is the frequency of y under μ. As a consequence of the ergodic theorem, $g_*\mu$ and μ are singular.

□

If g is trivial, or $p = (1/|X|, \ldots, 1/|X|)$, then $g_*\mu = \mu$. However, it is not sufficient for g to be nontrivial and $p \neq (1/|X|, \ldots, 1/|X|)$ for $g_*\mu$ and μ to be singular.

Example 5.4.13. *Let $X = \{1, 2, 3\}$, μ - a Bernoulli measure with probability vector $p = \{1/2, 1/4, 1/4\}$, and let g be generated by an automaton with a single state s with $\pi(s, x) = s$ for $x \in X$, $\lambda(g, 1) = 1$, $\lambda(g, 2) = 3$, $\lambda(g, 3) = 2$. Then $g_*\mu = \mu$.* △

This is a counter-example to Theorem 10 of [3], where the listed conditions were insufficient to guarantee singularity, and shows that additional conditions are needed.

In spite of Example 5.4.11, there are many cases when the vector t falls apart as a tensor product. In particular, we show the following:

Lemma 5.4.14. *Supppose that A is such that for any $s \in S$ and $x \in X$, there is $s' \in S$ such that $\pi(s', x) = s$. Then $k = \frac{1}{|S|}(1, 1, 1, \ldots, 1)$, and $t = k \otimes l$.*

Proof. the condition forces s' to be unique, by pigeonhole principle.

Therefore, if the condition holds, the columns of K sum to 1, and so $(1,\ldots,1)K = (1,1,\ldots,1)$. Normalizing by the sum, we obtain

$$k = \frac{1}{|S|}(1,1,\ldots,1).$$

To show the second part, note that the columns of T then have exactly $|X|$ nonzero entries: for any $x \in X$,$T_{(s',x)(s,y)} = L(x,y)$ only for the unique s' such that $\pi(s',x) = y$. Then, for all $s \in S$ and $y \in X$,

$$\begin{aligned}
|S|k \otimes l \cdot T_{_,(s,y)} &= (l_1, l_2, \ldots, l_{|X|}, \ldots, l_1, l_2, \ldots, l_{|X|}) \cdot T_{_,(s,y)} \\
&= \sum_{s' \in S} \sum_{x \in X} l_x \cdot \begin{cases} L(x,y), \text{ if } \pi(s',x) = y; \\ 0, \text{ otherwise} \end{cases} \\
&= \sum_{x \in X} l_x L(x,y) \\
&= l_y.
\end{aligned}$$

Therefore, $k \otimes l \cdot T_{\_,(s,y)} = \frac{1}{|S|} l_y$, and

$$k \otimes l \cdot T = \frac{1}{|S|}(l_1, l_2, \ldots, l_{|X|}, \ldots, l_1, l_2, \ldots, l_{|S|}) = k \otimes l.$$

Since

$$\sum_{s \in S} \sum_{x \in X} (k \otimes l)_{(s,x)} = \sum_{s \in S} \left(\sum_{x \in X} \frac{l_x}{S} \right) = \sum_{s \in S} \frac{1}{S} = 1,$$

$k \otimes l$ is the stationary vector of T, and the result follows. □

Definition 5.4.15: The automata for which the condition of Lemma 5.4.14 holds are called **reversive** automata.

We thus have the following

Corollary 5.4.16. *Suppose A is reversive, or μ is Bernoulli. Then $t = k \otimes l$.*

Since the proof of Theorem 5.4.12 follows from $t = k \otimes l$, we obtain a more general version for Markov measures:

Theorem 5.4.17. *Suppose A is reversive, or μ is Bernoulli. If $l \neq \frac{1}{|X|}(1, 1, \ldots, 1)$, then μ and $g_*\mu$ are singular.*

5.5 Examples on a 2-letter alphabet

Example 5.5.1. *In the examples that follow, $X = \{0, 1\}$, and L is in general form:*

$$L = \begin{bmatrix} 1-p & p \\ q & 1-q \end{bmatrix},$$

l is the stationary vector of L ($lL = L$ and $\sum l_i = 1$), so $l = \left(\frac{q}{p+q}, \frac{p}{p+q}\right)$, and μ is the Markov measure induced by L. K is a matrix $S \times S \to \mathbf{R}$ defined by

$$K_{s,s'} := \sum_{\pi(s,x)=s'} l(x)$$

and k is its stationary probability vector ($kK = k, \sum k_i = 1$). As before, T is given by

$$T_{(s_0,x_0)(s_1,x_1)} = \begin{cases} L(x_0, x_1), \text{if } \pi(s_0, x_0) = s_1; \\ 0, \text{otherwise}, \end{cases}$$

and t is its stationary probability vector ($tT = t, \sum t_i = 1$).

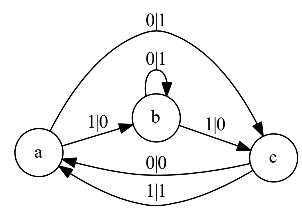
Finally, f is the vector of frequencies under the action of A (with any starting state):

$$f_x = \lim_{n \to \infty} \frac{1}{n} \sum_{m=0}^{n-1} \chi_{xX^{\mathbf{N}}} \sigma^m w$$

for μ-almost all w (σ is the shift).

5.5.1 The automaton F

This is a generalization of Example 5.4.9 to an arbitrary Markov chain on a 2-letter alphabet. This is a reversive automaton (in fact, bireversive: its dual is reversive), and so $t = k \otimes l$.



$$K = \begin{pmatrix} 0 & \frac{p}{p+q} & \frac{q}{p+q} \\ 0 & \frac{q}{p+q} & \frac{p}{p+q} \\ 1 & 0 & 0 \end{pmatrix}$$

$$k = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix}$$

$$T = \begin{pmatrix} 0 & 0 & 0 & 0 & 1-p & p \\ 0 & 0 & q & 1-q & 0 & 0 \\ 0 & 0 & 1-p & p & 0 & 0 \\ 0 & 0 & 0 & 0 & q & 1-q \\ 1-p & p & 0 & 0 & 0 & 0 \\ q & 1-q & 0 & 0 & 0 & 0 \end{pmatrix}$$

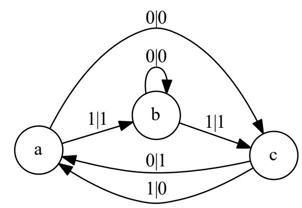
$$t = \begin{pmatrix} \frac{q}{3(p+q)} & \frac{p}{3(p+q)} & \frac{q}{3(p+q)} & \frac{p}{3(p+q)} & \frac{q}{3(p+q)} & \frac{p}{3(p+q)} \end{pmatrix}$$

$$k \otimes l = \begin{pmatrix} \frac{q}{3(p+q)} & \frac{p}{3(p+q)} & \frac{q}{3(p+q)} & \frac{p}{3(p+q)} & \frac{q}{3(p+q)} & \frac{p}{3(p+q)} \end{pmatrix}$$

$$f = \begin{pmatrix} \frac{2p+q}{3(p+q)} & \frac{p+2q}{3(p+q)} \end{pmatrix}$$

5.5.2 Bellaterra

Since the Bellaterra automaton is the product of $automatonF$ with an automaton that switches 0 and 1, the entries of f are switched, and the rest stays the same:



$$K = \begin{pmatrix} 0 & \frac{p}{p+q} & \frac{q}{p+q} \\ 0 & \frac{q}{p+q} & \frac{p}{p+q} \\ 1 & 0 & 0 \end{pmatrix}$$

$$k = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix}$$

$$T = \begin{pmatrix} 0 & 0 & 0 & 0 & 1-p & p \\ 0 & 0 & q & 1-q & 0 & 0 \\ 0 & 0 & 1-p & p & 0 & 0 \\ 0 & 0 & 0 & 0 & q & 1-q \\ 1-p & p & 0 & 0 & 0 & 0 \\ q & 1-q & 0 & 0 & 0 & 0 \end{pmatrix}$$

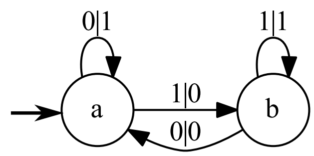
$$t = \begin{pmatrix} \frac{q}{3(p+q)} & \frac{p}{3(p+q)} & \frac{q}{3(p+q)} & \frac{p}{3(p+q)} & \frac{q}{3(p+q)} & \frac{p}{3(p+q)} \end{pmatrix}$$

$$k \otimes l = \begin{pmatrix} \frac{q}{3(p+q)} & \frac{p}{3(p+q)} & \frac{q}{3(p+q)} & \frac{p}{3(p+q)} & \frac{q}{3(p+q)} & \frac{p}{3(p+q)} \end{pmatrix}$$

$$f = \begin{pmatrix} \frac{p+2q}{3(p+q)} & \frac{2p+q}{3(p+q)} \end{pmatrix}$$

5.5.3 Lamplighter

The Lamplighter automaton is interesting in that the output frequencies don't depend on the input frequencies. This is again a reversive automaton.



$$K = \begin{pmatrix} \frac{q}{p+q} & \frac{p}{p+q} \\ \frac{p}{p+q} & \frac{q}{p+q} \end{pmatrix}$$

$$k = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

$$T = \begin{pmatrix} 1-p & p & 0 & 0 \\ 0 & 0 & q & 1-q \\ 0 & 0 & 1-p & p \\ q & 1-q & 0 & 0 \end{pmatrix}$$

$$t = \begin{pmatrix} \frac{q}{2(p+q)} & \frac{p}{2(p+q)} & \frac{q}{2(p+q)} & \frac{p}{2(p+q)} \end{pmatrix}$$

$$k \otimes l = \begin{pmatrix} \frac{q}{2(p+q)} & \frac{p}{2(p+q)} & \frac{q}{2(p+q)} & \frac{p}{2(p+q)} \end{pmatrix}$$

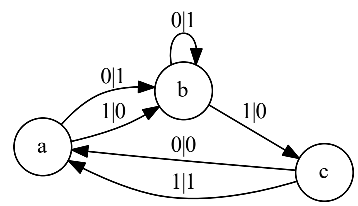
$$f = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

Note: even though the frequencies of individual characters don't depend on p and q, this is not the case for words of length 2. The input and output frequencies are as follows:

Word:	00	01	10	11
Input frequency:	$\frac{q-pq}{p+q}$	$\frac{pq}{p+q}$	$\frac{pq}{p+q}$	$\frac{p-pq}{p+q}$
Output frequency:	$\frac{q}{2(p+q)}$	$\frac{p}{2(p+q)}$	$\frac{p}{2(p+q)}$	$\frac{q}{2(p+q)}$

5.5.4 Case when $t \neq k \otimes l$

This can happen even with a 2-character alphabet. This automaton differs from Aleshin only by one arrow, but that change made the automaton non-reversive.



$$K = \begin{pmatrix} 0 & 1 & 0 \\ 0 & \frac{q}{p+q} & \frac{p}{p+q} \\ 1 & 0 & 0 \end{pmatrix}$$

$$k = \begin{pmatrix} \frac{p}{3p+q} & \frac{p+q}{3p+q} & \frac{p}{3p+q} \end{pmatrix}$$

$$T = \begin{pmatrix} 0 & 0 & 1-p & p & 0 & 0 \\ 0 & 0 & q & 1-q & 0 & 0 \\ 0 & 0 & 1-p & p & 0 & 0 \\ 0 & 0 & 0 & 0 & q & 1-q \\ 1-p & p & 0 & 0 & 0 & 0 \\ q & 1-q & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$t = \Big(-\frac{pq(p+q-2)}{qp^2+(2q^2-3q+3)p+(q^2-3q+3)q} \quad \frac{p\left(q^2+(p-2)q+1\right)}{qp^2+(2q^2-3q+3)p+(q^2-3q+3)q} \quad \frac{q\left(p^2+(2q-3)p+q^2-3q+3\right)}{qp^2+(2q^2-3q+3)p+(q^2-3q+3)q}$$
$$\frac{p}{qp^2+(2q^2-3q+3)p+(q^2-3q+3)q} \quad \frac{pq}{qp^2+(2q^2-3q+3)p+(q^2-3q+3)q} \quad \frac{p-pq}{qp^2+(2q^2-3q+3)p+(q^2-3q+3)q} \Big)$$

$$k \otimes l = \begin{pmatrix} \frac{pq}{(p+q)(3p+q)} & \frac{p^2}{(p+q)(3p+q)} & \frac{q}{3p+q} & \frac{p}{3p+q} & \frac{pq}{(p+q)(3p+q)} & \frac{p^2}{(p+q)(3p+q)} \end{pmatrix}$$

$$f = \begin{pmatrix} \frac{p\left(q^2+(p-1)q+2\right)}{qp^2+(2q^2-3q+3)p+(q^2-3q+3)q} & \frac{p(q-1)^2+\left(q^2-3q+3\right)q}{qp^2+(2q^2-3q+3)p+(q^2-3q+3)q} \end{pmatrix}$$

5.6 Reversive automata

Recall that an automaton is **reversive** (Def. 5.4.1) if it satisfies the condition of Lemma 5.4.14: if one can arrive to any state by any letter of the alphabet. Equivalently, that means that X acts on the states of the automaton as a group (by $x \cdot g = \pi(g, x)$).

An immediate corollary of Theorem 5.4.7 and Lemma 5.4.14 is the following

Corollary 5.6.1. *Let g, A, k, l, μ, etc. be as before. If A is reversive, then for all $x \in X$ and almost all $w \in X^{\mathbf{N}}$:*

$$\lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \chi_x \sigma^n(gw) = \sum_{s_0 \xrightarrow[x]{y_0} s_1} k(s_0) l(y_0).$$

These automata act on bi-infinite sequences $w : \mathbf{Z} \to X$. To define the action, for a reversive automaton A, let $\hat{A}$ be the automaton obtained by reversing the arrows in A, i.e.

$$\pi_A(s, x) = t \Leftrightarrow \pi_{\hat{A}}(t, x) = s$$

$$\lambda_A(s, x) = y \Leftrightarrow \lambda_{\hat{A}}(\pi(s, x), x) = y.$$

This is well-defined when A is reversive; we call $\hat{A}$ the reverse of A.

As before, we can extend $\pi_{\hat{A}}$, $\lambda_{\hat{A}}$ to $X^{-\mathbf{N}}$: for $w \in X^{-\mathbf{N}}$ and $x \in X$, define

$$\hat{\pi}(s, wx) = \hat{\pi}(\pi_{\hat{A}}(s, x), w);$$

$$\hat{\lambda}(s, wx) = \hat{\lambda}(\pi_{\hat{A}}(s, x), w).$$

Now we can extend $\pi = \pi_A$ to $w \in X^{\mathbf{Z}}$: for $w = \ldots w_{-1} w_0 w_1 \ldots$, set

$$\pi(w) := \hat{\pi}(\ldots w_{-2} w_{-1} w_0) \pi(w_0 w_1 w_2 \ldots).$$

Theorem 5.4.7 holds in this setting as well, after being reformulated for bi-infinite sequences:

Theorem 5.6.2. *If A is reversive, then*

$$\lim_{n\to\infty} \frac{1}{2n+1}\left(\sum_{k=1}^{n} \chi_x \sigma^n(gw) + \sum_{k=1}^{n} \chi_x \sigma^{-n}(gw)\right) = \sum_{s_0 \xrightarrow[x]{y_0} s_1} k(s_0) l(y_0).$$

5.7 When the pushforward is Gibbsian

A *sofic measure* is an image of a Markov measure under a projection. Thus, by definition, the measure Q is sofic.

It is interesting to consider when Q belongs to a class that lies between sofic and Markovian measures: the so-called Bowen-Gibbs measures. Recall (from definition 2.2.1) that these are measures ν such that for some $C, P \in \mathbf{R}$ ($C > 0$) and all $w \in X^n n$,

$$\frac{1}{C} < \frac{\nu(w_0 \dots w_{n-1} X^{\mathbf{N}})}{exp(-nP + \sum_{i=0}^{n-1} f(\sigma^i w))} < C.$$

Chazottes and Ugalde have provided in [9] a sufficient condition for an image of a Markov measure under a factor map to be Gibbsian (we follow a rephrasing in [10]):

Theorem 5.7.1 (Chazotte, Ugalde)**.** *Let A and B be mixing 1-step one-sided shifts of finite type. Let $\phi : A \to B$ be a one-block factor map with the following properties:*

- *for a 2-block $b_1 b_2$ in $B \times B$, a letter $a_1 \in A$ such that $\phi(a_1) = b_1$ can be extended to a two-block $a_1 a_2$ in $X \times X$ such that $\phi(a_1 a_2) = b_1 b_2$;*
- *given a periodic point $b \in B$ with period m less than or equal to the number of letters appearing in B, any pair of letters a_0, a_{m-1} mapping to b_0, b_{m-1}, respectively, can be extended to a word $a_0, \dots a_{m-1}$ of length m that maps to $b_0, \dots, b_{m-1}$.*

Then $\phi_*\mu$ is a Gibbs measure.

In our setup, A is the subshift of $(S \times X)^{\mathbf{N}}$ defined by nonzero entries of T, B is the full shift $X^{\mathbf{N}}$, and $\phi = \tilde{\lambda}$.

Since B is the full shfit, B is 1-step and mixing.

A is mixing whenever the automaton $\mathcal{A}$ is μ-strongly-connected.

The map $\tilde{\lambda} : (S \times X)^{\mathbf{N}} \to X^{\mathbf{N}}$ is induced by the map λ, which is as a map from the alphabet $S \times X$ to alphabet X. Therefore, $\tilde{\lambda}$ is, in fact, a 1-block factor map.

The first condition translates to the following: for every sequence $y_1 y_2$, and for every (s_1, x_1) such that $\lambda(s_1, x_1) = y_1$, there exists (s_2, x_2) such that:

- $\pi(s_1, x_1) = s_2$;
- $\lambda(s_2, x_2) = y_2$;
- $L(x_1, x_2) > 0$.

Note that the first two conditions are satisfied whenever the automaton A is invertible by setting $s_2 = \pi(s_1, x_1)$ and $x_2 = \lambda_{s_2}^{-1}(y_2)$. The addition of the third condition is satisfied by making A μ-strongly-connected.

The second condition of is satisfied trivially for binary alphabets, i.e. when $|X| = 2$.

The sufficient conditions of [9] translate to the following corollary:

Corollary 5.7.2. *Let X be a binary alphabet, and let A be invertible, μ-strongly connected automaton.*

Then Q is Gibbsian.

This condition is not satisfied for non-invertible automata.

Let $X = 0, 1$. Consider an automaton A with four states, A, B, C, D, which is a Moore machine (i.e. its output only depends on the current state), where A, B output 0, and C, D output 1, and the transition is given by $A \to_1 B \to_0 C \to_0 D \to_0 A$.

Then the matrix in the condition of Chazottes and Ugalde has zero rows.

In [10], Yoo has shown that if $h : X^\infty \to Y^\infty$ is a 1-block factor map (i.e. induced by a map $h : X \to Y$), a sufficient condition for $h_*\mu$ to be Gibbsian is the following:

Theorem 5.7.3 (Yoo)**.** *If there is a k such that for every pair $u, v \in X^k$ satisfying $h(u) = h(v)$, there is $w \in X^k$ such that $h(u) = h(v) = h(w)$, with $w_1 = u_1$ and $w_k = v_k$, then $h_*\mu$ is Gibbsian.*

In our case, we have the map $\tilde{\lambda} : (S \times X)^{\mathbf{N}} \to X^\infty$. The the condition becomes the following: for every two paths with the same output:

$$s_1 \xrightarrow[y_1]{x_1} s_2 \ldots s_n \xrightarrow[y_n]{x_n}$$
$$\hat{s}_1 \xrightarrow[y_1]{\hat{x}_1} s_2 \ldots \hat{s}_n \xrightarrow[y_n]{\hat{x}_n}$$

there must exist a path

$$s'_1 \xrightarrow[y_1]{x'_1} s'_1 \ldots s'_n \xrightarrow[y_n]{x'_n}$$

with $(s'_1, x'_1) = (s_1, x_1)$, and $(s'_n, x'_n) = (\hat{s}_n, \hat{x}_n)$.

If A is invertible, two paths from the same initial state differ in the first output character already. The only way the condition can be satisfied is if all paths of length n that give the same output end up in the same state (for some n).

6. CONCLUSIONS AND OPEN PROBLEMS

Finite-state measures are an interesting class that naturally generalizes Markov measures in a different way than Gibbs measures do (a finite-state measure may, but need not be Gibbs). The connections between automata, finite-state, and Gibbs measures lead to the following open questions:

1. Are there non-unifixed finite-state measures that are still Gibbs?
2. What can the measure μ induced by the automatic logarithm tell us about the structure of the Schreier graph? What specifically can be said in the cases when it is finite state?
3. What can be said about the images of Gibbs measures under the action of finite automata?
4. Describe the images of finite-state measures under the action of non-invertible tree endomorphisms;
5. Study the relationship between quasi-finite-state measures and other classes of measures.

REFERENCES

[1] V. Kudryavtsev, S. Aleshin, and A. Podkolzin, *Vvedeniye v Teoriyu Avtomatov [Introduction to Automata Theory].* Nauka, Moscow, 1985.

[2] A. Ryabinin, “Stochastic functions of finite automata,” *Algebra, Logika i Teoriya Chisel*, vol. 24, pp. 77–80, 1986.

[3] R. Grigorchuk and R. Kravchenko, “On the lattice of subgroups of the lamplighter group,” *Internat. J. Algebra Comput.*, vol. 24, no. 6, pp. 837–877, 2014.

[4] L. Sholomov, *Osnovy teorii diskretnykh logicheskikh i vychislitel’nykh ustroistv*. Moscow: Nauka, 1980.

[5] R. I. Grigorchuk, V. V. Nekrashevich, and V. I. Sushchanskiĭ, “Automata, dynamical systems, and groups,” *Tr. Mat. Inst. Steklova*, vol. 231, no. Din. Sist., Avtom. i Beskon. Gruppy, pp. 134–214, 2000.

[6] R. Bowen, D. Ruelle, and J. Chazottes, *Equilibrium States and the Ergodic Theory of Anosov Diffeomorphisms*. Lecture Notes in Mathematics, Springer Berlin Heidelberg, 2008.

[7] P. Billingsley, *Ergodic Theory and Information*. Wiley series in probability and mathematical statistics, J. Wiley, 1965.

[8] H. Bass, M. V. Otero-Espinar, D. Rockmore, and C. Tresser, *Cyclic renormalization and automorphism groups of rooted trees*, vol. 1621 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1996.

[9] J.-R. Chazottes and E. Ugalde, “Projection of markov measures may be gibbsian,” *Journal of Statistical Physics*, vol. 111, pp. 1245–1272, Jun 2003.

[10] J. Yoo, “On Factor Maps that Send Markov Measures to Gibbs Measures,” *Journal of Statistical Physics*, vol. 141, pp. 1055–1070, Dec. 2010.

APPENDIX A

MEALY AUTOMATA PRODUCT AND MINIMIZATION

Here we present a Java class to represent Mealy automata, along with methods to obtain products, inverse, and minimization.

The implementation is for input alphabet X coinciding with output alphabet Y, but can be easily adapted for the case when they differ, as the algorithms stay the same.

```
public class MealyAutomaton {
    /**
     * Number of states of this automaton
     */
    int size;

    /**
     * Input alphabet. Shouldn't overlap.
     */
    String[] alphabet;
    String [] labels;
    int[][] transition;
    int[][] output;
    String name;

    public MealyAutomaton(String name, String[] alphabet, String[]
        labels,
            int[][] transition, int[][] output) {
        this.name = name;
        this.alphabet = alphabet;
        this.labels = labels;
```

```java
        this.transition = transition;
        this.output = output;
        this.size = transition.length;
    }


    public static int[] actOn(MealyAutomaton M, int start_state, int[]
        input)
    {
        int[] ans = new int[input.length];
        //System.out.print(M.labels[start_state]);
        for (int i=0; i<input.length; i++){
            ans[i] = M.output[start_state][input[i]];
            start_state = M.transition[start_state][input[i]];
            //System.out.print(M.labels[start_state]);
        }
        //System.out.println();
        return ans;
    }


    static String cat(String a, String b){
        //return a + b;
        return (a.equals("I")) ? b : (b.equals("I"))? a : a+b;
    }

    static MealyAutomaton multiply(MealyAutomaton A, MealyAutomaton B){
        if (!Arrays.equals(A.alphabet, B.alphabet)) { return null;}
        else {
            String name = A.name +"_x_" + B.name;
            String[] alphabet= A.alphabet;
```

```
        int size = A.size * B.size;
        String[] labels = new String[size];
        int[][] transition = new int[size][];
        int[][] output = new int[size][];

        for (int i=0; i<A.size; i++){
            for (int j=0; j<B.size; j++){
                int k = i*B.size + j;
                output[k] = new int[alphabet.length];
                transition[k] = new int[alphabet.length];
                for (int c=0; c<alphabet.length; c++){
                    output[k][c] = A.output[i][B.output[j][c]];
                    int j2 = B.transition[j][c];
                    int i2 = A.transition[i][B.output[j][c]];
                    transition[k][c] = i2*B.size + j2;
                }
                labels[k] = cat(A.labels[i], B.labels[j]);
            }
        }
        return new MealyAutomaton(name, alphabet, labels,
            transition, output);
    }
}

static MealyAutomaton inverse(MealyAutomaton A){
        String[] alphabet= A.alphabet;
        int size = A.size;
        String[] labels = new String[size];
        int[][] transition = new int[size][];
        int[][] output = new int[size][];
```

```
        for (int i=0; i<A.size; i++){
            output[i] = new int[alphabet.length];
            transition[i] = new int[alphabet.length];
            for (int c=0; c<alphabet.length; c++)
            {
                transition[i][A.output[i][c]] =
                    A.transition[i][c];
                output[i][A.output[i][c]] = c;
            }
            labels[i] = A.labels[i]+"_";
            //labels[i] = A.labels[i]+"_inv";
            //labels[i] = A.labels[i]+"*";
        }
        return new MealyAutomaton(alphabet, labels, transition,
            output);
    }

    public MealyAutomaton minimize(){
        int[] color = new int[size];
        for (int i=0; i<size; i++) {color[i] = 0;}

        final int MAX_COLS = size*2; //we might have up to 2 x size
            colors reserved, but no more

        int[] succ_color = new int[MAX_COLS];
        Arrays.fill(succ_color, -1);
        succ_color[0] = 1;

        int color_count = 1;
        int next_free_color = 2;
```

```
int[] reps = new int[MAX_COLS];   //again, color labels might
    go as high as 2 x size
Arrays.fill(reps, -1);
reps[0]=0;

boolean[] color_split = new boolean[MAX_COLS];
boolean[] tagged_for_recolor = new boolean[size];

boolean mismatch_found = true;
while(mismatch_found){
mismatch_found = false;
     for (int i=0; i<size; i++){
          int rep = reps[color[i]];
          boolean in_right_class = true;
          int c;
          for (c = 0; in_right_class && (c < alphabet.length);
              c++){
               in_right_class =  in_right_class
                              && (output[rep][c] == output[i][c])
                              && (color[transition[rep][c]] ==
                                  color[transition[i][c]]);
          }
          if (!in_right_class){
               tagged_for_recolor[i] = true;
          }
     }
     //recolor the mismatched nodes

     for (int i=0; i<size; i++){
          if (tagged_for_recolor[i]){
               tagged_for_recolor[i] = false;
```

```
            color_split[color[i]] = true;
            color[i] = succ_color[color[i]];
            reps[color[i]] = i;
        }
    }

    //fix successor colors and update color count
    for (int i=0; i< MAX_COLS; i++){
        if (color_split[i]){
            succ_color[succ_color[i]] = next_free_color;
            next_free_color ++;

            succ_color[i] = next_free_color;
            next_free_color++;
            color_split[i] = false;
            mismatch_found = true;
            color_count++;
        }
    }
}

//first, group used colors into an array, and build a reverse
   index
int[] used_colors = new int[color_count]; //store all colors
   used in the graph,
Arrays.fill(used_colors, -1);              //i.e. such that
   there's at leat one node of that color
int[] color_index = new int[MAX_COLS];     //reverse index:
   given color, give its index
Arrays.fill(color_index, -1);              // in the
   used_colors array
```

```java
        int caret = 0;
        for (int i=0; i < MAX_COLS; i++){
            if (reps[i] > -1){
                used_colors[caret] = i;
                color_index[i] = caret;
                caret++;
            }
        }

        //now build labels, transition and output (the alphabet is the
            same)
        String[] new_labels = new String [color_count];
        int[][] new_output = new int[color_count][];
        int[][] new_transition = new int[color_count][];
        for (int i=0; i<used_colors.length; i++) {
            new_output[i] = new int[alphabet.length];
            new_transition[i] = new int[alphabet.length];
            for (int c = 0; c < alphabet.length; c++){
                new_output[i][c] = output[reps[used_colors[i]]][c];
                new_transition[i][c] =
                    color_index[color[transition[reps[used_colors[i]]][c]]];
            }
            new_labels[i] = labels[reps[used_colors[i]]];
        }

        return new MealyAutomaton(alphabet, new_labels,
            new_transition, new_output);
    }
}
```