PROBABILISTIC AUTOMATIC COMPLEXITY OF FINITE STRINGS

KENNETH GILL

Abstract. We introduce a new complexity measure for finite strings using probabilistic finite-state automata (PFAs), in the same spirit as existing notions employing DFAs and NFAs, and explore its properties. The PFA complexity $A_P(x)$ is the least number of states of a PFA for which x is the most likely string of its length to be accepted. The variant $A_{P,\delta}(x)$ adds a real-valued parameter δ specifying a lower bound on the gap in acceptance probabilities between x and other strings. We relate A_P to the DFA and NFA complexities, obtain a complete classification of binary strings with $A_P = 2$, and prove $A_{P,\delta}(x)$ is computable for every x and for cofinitely many δ (depending on x). Finally, we discuss several other variations on A_P with a view to obtaining additional desirable properties.

1. Introduction

Informally, the Kolmogorov complexity of a finite string w is the size of the smallest Turing machine which outputs w given no input. As a function, it is well-known to be noncomputable, and moreover only defined up to an additive constant. These drawbacks have motivated several authors to define complexity measures based on models of computation less powerful than the Turing machine, such as context-free grammars [8, 1]. In 2001, Shallit and Wang introduced one such measure using deterministic finite-state automata (DFAs), defining $A_D(w)$ to be the number of states of the smallest DFA for which w is the only string of its length to be accepted [23]. A_D is computable, well-defined, and there is a polynomial-time algorithm to recover w from a witness for $A_D(w)$. Later in 2013, Hyde defined a similar measure A_N replacing DFAs with nondeterministic finite-state automata (NFAs) [12, 13]. A_N shares the advantages of A_D over Kolmogorov complexity while additionally making $A_N(w) = A_N(\overleftarrow{w})$, where $\overleftarrow{w}$ is the reversal of w, and avoiding "dead states" (nonaccepting states with no out-transitions) often present among witnesses for A_D merely to satisfy the requirement of determinism. The study of A_N has been continued by Kjos-Hanssen, see e.g. [14, 16, 17, 18], as well as the recent book [15].

Inspired by the aforementioned work, we investigate what happens when deterministic or nondeterministic machines are replaced by probabilistic ones (PFAs), wherein each state transition occurs with some probability and each word w is

Date: February 14, 2024.

Most of this work is part of the author's Ph.D. dissertation at Penn State University [11]. I would like to thank my advisors Linda Westrick and Jan Reimann for their invaluable help and support, and also Jake Canel for helpful conversations leading to the proof of Theorem 5.1. This research was supported in part by NSF grant DMS-1854107.

assigned a probability of acceptance $\rho_M(w)$ by the PFA M. We view M as describing w if $\rho_M(w)$ is uniquely maximized among all strings of length $|w|$, and let the PFA complexity be the minimal number of states of such an M:

Definition 1.1. The *probabilistic automatic complexity* (PFA complexity) of w is

$$A_P(w) = \min\{\, k \;:\; \text{there is a } k\text{-state PFA } M \text{ such that } \operatorname{gap}_M(w) > 0 \,\},$$

where

$$\operatorname{gap}_M(w) = \min\{\, \rho_M(w) - \rho_M(z) : |z| = |w| \text{ and } z \neq w \,\}.$$

Here all words are presumed to be drawn from some finite alphabet Σ fixed in advance. This definition is probably the one most directly analogous to the definitions of A_D and A_N. Our main result about A_P is the following complete classification of binary strings with complexity 2, which demonstrates that it does appear to capture some intuitive structural properties of strings:

Theorem 4.1. *For a binary string w, $A_P(w) = 2$ if and only if w is of the form*

$$i^n j^m, \qquad i^n j^m i, \qquad i^n (ji)^m, \qquad \text{or} \quad i^n j (ij)^m$$

for some $n, m \geq 0$, where $i, j \in \{0, 1\}$.

One can compare this to the fact that $A_N(w) = 2$ if and only if $w = ij^n$, $i^n j$, or $(ij)^n$ [12]. Indeed, A_N is unbounded on strings of the form $i^n j^m$, which gives us

Corollary 4.13. *$A_N(w) - A_P(w)$ may be arbitrarily large among binary w.*

On the other hand, while the DFA and NFA complexities are computable simply by virtue of there being only finitely many DFAs and NFAs of a given number of states over a given alphabet, the computability of A_P is not at all evident from the definition, and remains open at the time of writing. It follows from Theorem 4.1 that in passing to A_P, we once more lose the property that the complexity of a string equals that of its reversal. And there are nonconstant strings w such that $A_P(w)$ is witnessed by a PFA with dead states, as with A_D. (There is no w such that $A_P(w)$ *must* be witnessed by such a PFA, though, because $\rho_M(w)$ is continuous in the transition probabilities of M and one can always perturb these probabilities so that all are positive.)

Another natural concern one may have with A_P is that M can witness $A_P(w)$ while $\rho_M(w)$ is not very high, or not much different from $\rho_M(z)$ for other strings z of the same length. Is M really a good representation of w if it can only slightly distinguish w from other strings? What if all potential witnesses M have this property? We attempt to define our way out of this problem by introducing a real-valued parameter giving a lower bound on the gap between probabilities:

Definition 1.2. The *probabilistic automatic complexity of w with gap $\delta \in [0, 1)$* is[1]

$$A_{P,\delta}(w) = \min\{\, k \;:\; \text{there is a } k\text{-state PFA } M \text{ such that } \operatorname{gap}_M(w) > \delta \,\}.$$

Thus $A_{P,0}(w) = A_P(w)$. It turns out that $A_{P,\delta}(w)$ is computable for all w and almost all $\delta \neq 0$, which is the other main result of this paper.

[1] This is a slightly different definition from that originally given by the author in [11], which required $\operatorname{gap}_M(w) \geq \delta$ rather than $>$. The author has come to feel that the present definition is more natural. Only minor amendments to the proofs of results involving $A_{P,\delta}$ were needed as a result of this change.

Theorem 5.1. *For any finite alphabet Σ, the function $(\delta, w) \mapsto A_{P,\delta}(w)$ is*

- *Continuous everywhere on $[0,1) \times \Sigma^*$ except on a countably infinite set which can be enumerated by a single algorithm;*
- *Computable on $(0,1) \times \Sigma^*$ where it is continuous.*

In particular, for every w, $A_{P,\delta}(w)$ is computable for all but at most $A_D(w) - 2$ many values of δ, and is continuous at $\delta = 0$.

The reader should note that this theorem makes no positive or negative claim about computability at $\delta = 0$.

$A_{P,\delta}$ has another philosophically attractive feature which we now describe. Suppose one is given an automaton M as a "black box", that is, with no information whatsoever about its inner workings. All one can do is run it with some input string, and check whether it accepts or rejects the string. Suppose further that an experimenter wishes to test whether this automaton witnesses an upper bound for $A_P(w)$ for some string w. Then the experimenter needs not only to check whether each $z \in \Sigma^{|w|}$ is accepted, but whether or not it will be accepted with a lower bound λ on its probability of acceptance, for each λ in turn. (This would enable them to decide if there is some particular w, λ with $\rho_M(w) > \lambda$ but $\rho_M(z) < \lambda$ whenever $|z| = |w|$ and $z \neq w$. In other words, they would estimate a lower bound on $\text{gap}_M(w)$.) The experimenter can only accomplish this by running the machine repeatedly on each input w to get some sense of the expected value of $\rho_M(w)$, up to some acceptable margin of error ε.

In his original paper introducing PFAs, Rabin [21] discusses a similar endeavor in the context of establishing experimentally that w is in a given stochastic language, where a language is stochastic if it is of the form $\{\, w \in \Sigma^* : \rho_M(w) > \lambda \,\}$ for some PFA M and $\lambda \in [0,1]$, called the *cut-point*. As he points out, the law of large numbers implies that as long as $\rho_M(w) \neq \lambda$, there is a finite number $N = N(w, \varepsilon)$ such that running N trials, counting the number s of successes, and comparing s/N with λ will correctly determine if $\rho_M(w) > \lambda$ with probability $1 - \varepsilon$. But, as he goes on to say, finding $N(w, \varepsilon)$ would depend on knowing $\rho_M(w)$ in the first place.

Rabin's solution is to only consider cut-points λ which are isolated for M, that is, such that $|\rho_M(w) - \lambda| \geq \delta$ for all $w \in \Sigma^*$ and some $\delta > 0$. If one wants to run the above experiment to test if $\rho_M(w) > \lambda$ when λ is isolated, then the number of trials N needed to determine this within margin of error ε now only depends on δ and ε—regardless of M. Knowledge of $\rho_M(w)$ is not needed. Of course, this is not a solution from a practical point of view if no such cut-point is given at the outset, because now the experimenter would need to determine if λ is isolated for M and (if so) a lower bound for its degree of isolation δ. The problem of determining if a given rational cut-point is isolated for a given PFA is known to be Σ^0_2-complete [5, Theorem 1].

But—back to our black-box experiment—if one specifies δ at the outset and looks for a witness for an upper bound on $A_{P,\delta}(w)$ rather than $A_P(w)$, the problem disappears and we still get that N depends only on δ and ε, with both of these parameters now being chosen by the experimenter. To see why, let a single trial consist of running every word of length $|w|$ through the machine M once. If $s(w, N)$ is the number of acceptances of w in N trials, then there is a function $N = N(\delta, \varepsilon)$ such that for each $z \in \Sigma^{|w|}$, one correctly concludes with probability at least $1-\varepsilon'$ that $\rho_M(w)-\rho_M(z)>\delta$ given $[s(w,N)-s(z,N)]/N>\delta$, assuming $\rho_M(w)-\rho_M(z)\neq\delta$. Here ε' is chosen small enough that $(1-\varepsilon')^{|\Sigma|^{|w|}}>1-\varepsilon$. Since acceptances or rejections of words are presumed to be independent events, it follows that after $N(\delta,\varepsilon)$ trials, one correctly concludes with probability at least $1-\varepsilon$ that $\mathrm{gap}_M(w)>\delta$.

The structure of the rest of the paper is as follows. In the next section we collect some formal definitions needed later. In Section 3 are a few preliminary results, including Proposition 3.1 which relates A_P and $A_{P,\delta}$ to A_D and A_N. Section 4, which takes up over half of the paper, consists entirely of the proof of Theorem 4.1. A correspondence between PFAs and iterated function systems is discussed in Section 4.1, and this correspondence is used to prove both directions of the theorem in Sections 4.2 and 4.3. Some corollaries and additional remarks appear in Section 4.4. Then Section 5 is devoted to the proof of Theorem 5.1. Finally, in Section 6 we discuss several further variations on A_P with an eye to mitigating its potential flaws as a complexity measure.

2. Background

Let Σ^* be the set of finite strings over the finite alphabet Σ. Write $x^\frown y$ for the concatenation of the strings x and y.

Definition 2.1. A *probabilistic finite-state automaton* (PFA) is an abstract machine specified by a tuple $M=(S,\Sigma,P,\vec{\pi},\vec{\eta})$, where

- $S=\{\,s_1,\dots,s_n\,\}$ is the set of states;
- Σ is a finite alphabet;
- P is a set of $n\times n$ right-stochastic matrices $\{\,P_\sigma:\sigma\in\Sigma\,\}$ describing the transition probabilities. For each $\sigma\in\Sigma$, $(P_\sigma)_{ij}$ is the probability of going from s_i to s_j when letter σ is read;
- $\vec{\pi}$ is a row vector of length n giving a probability distribution on initial states, so $\vec{\pi}_j$ is the probability of the machine starting in state s_j; and
- $\vec{\eta}$ is a column vector of length n giving the list of accepting states, so $\vec{\eta}_i$ is 1 if s_i is accepting and is 0 otherwise.

When Σ is not important, we will frequently omit its mention, and likewise we usually identify S with the set $\{1,\dots,n\}$ for some n. Thus one can fully define a PFA by $\vec{\pi}$, $\vec{\eta}$, and the matrices P_σ. If all entries of $\vec{\pi}$ and each P_σ are rational numbers, then we refer to M as *rational.* PFAs can also be represented as digraphs, with edges labeled by transition probabilities. For example, Figure 1 depicts the PFA over the alphabet $\{0,1\}$ with

$$\vec{\pi}=(1,0,0),\quad \vec{\eta}=\begin{pmatrix}0\\0\\1\end{pmatrix},\quad P_0=\begin{pmatrix}0&0&1\\1&0&0\\1&0&0\end{pmatrix},\quad P_1=\begin{pmatrix}.5&0&.5\\0&1&0\\.5&.5&0\end{pmatrix}.$$

Definition 2.2. If M is a PFA and $x=x_1x_2\cdots x_\ell$ is a string, let

$$P_M(x)=P_{x_1}P_{x_2}\cdots P_{x_\ell}.$$

Then the *acceptance probability of x* with respect to M is

$$\rho_M(x)=\vec{\pi}P_M(x)\vec{\eta}.$$

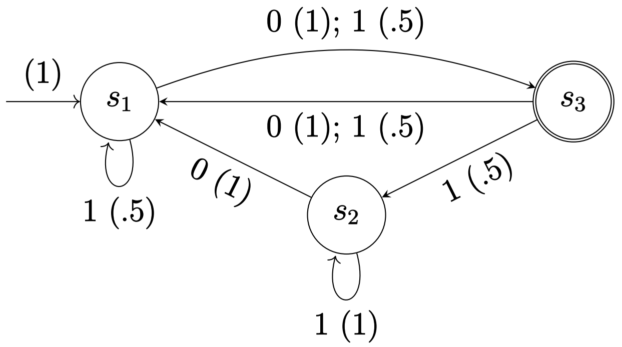


FIGURE 1. An example of a PFA. Numbers in parentheses are transition probabilities, so that the PFA starts in state s_1 with probability 1. s_3 is the unique accepting state.

If M is understood from context we may simply write $\rho(x)$, and similarly $\text{gap}(x)$.

One can view a DFA as the special case of a PFA in which $\vec{\pi}$ is a coordinate vector and all P_σs are permutation matrices. An NFA is then a slight relaxation of a DFA where $\vec{\pi}$ may be any zero-one vector and each P_σ may be any zero-one matrix. Of course, DFAs and NFAs are usually represented as digraphs, but it is convenient for us to think of them via their transition matrices since we will manipulate them directly alongside PFAs. The precise definitions of the DFA and NFA complexities are as follows:

Definition 2.3 (Shallit and Wang [23]). The *deterministic automatic complexity* of a finite string x is

$$A_D(x) = \min\{k : \text{there is a } k\text{-state DFA accepting } x \\ \text{uniquely among strings of length } |x|\}.$$

In other words, thinking of a witnessing DFA M as a PFA, this says $\text{gap}_M(x) = 1$, or equivalently $\text{gap}_M(x) > 0$ since the gap function takes only the values 0 and 1 when M is deterministic. It follows immediately that $A_P(x) \leq A_D(x)$ for all x.

Definition 2.4 (Hyde [12]). The *nondeterministic automatic complexity* of x is

$$A_N(x) = \min\{k : \text{there is a } k\text{-state NFA accepting } x \\ \text{and with a unique accepting path of length } |x|\}.$$

Every DFA witnessing $A_D(x)$ is an NFA that accepts x with a unique accepting path of length $|x|$ (by virtue of its determinism), so $A_N(x) \leq A_D(x)$ for all x.

For the reader's convenience, we repeat here the definitions of the gap function, A_P, and $A_{P,\delta}$.

Definition 1.1. The *probabilistic automatic complexity* (PFA complexity) of w is

$$A_P(w) = \min\{\, k \;:\; \text{there is a } k\text{-state PFA } M \text{ such that } \text{gap}_M(w) > 0\,\},$$

where

$$\text{gap}_M(w) = \min\{\, \rho_M(w) - \rho_M(z) : |z| = |w| \text{ and } z \neq w\,\}.$$

Definition 1.2. The *probabilistic automatic complexity of w with gap $\delta \in [0,1)$* is

$$A_{P,\delta}(w) = \min\{\, k \;:\; \text{there is a } k\text{-state PFA } M \text{ such that } \text{gap}_M(w) > \delta\,\}.$$

PFAs were independently introduced in 1963 by Michael Rabin and J. W. Carlyle [21, 4]. Carlyle's stochastic sequential machines are transducers with both input and output behavior, while Rabin's PFAs—which are sometimes also called stochastic acceptors—can only accept an input string with some probability. The present work focuses only on PFAs as defined by Rabin, although Carlyle-style machines have found wide applicability in machine learning and pattern recognition; see [26] for a modern survey. A notion of transducer complexity of finite strings has also been studied [3], but the approach taken there is most directly analogous to that of the Kolmogorov complexity rather than A_D. We leave the probabilistic analogue for future work. There is, however, an idea related to A_P which has been studied for transducers in the machine learning literature. Given a probabilistic finite-state transducer T, x is called the *most probable string* or *consensus string* of T if it is generated by T with maximal probability among all strings, not just among those with the same length [26]. One might ask if this notion should be adapted to PFAs, defining the complexity of x instead as the smallest size (in some sense) of a PFA accepting x with unique highest probability among all strings. But we will see in the proof of Theorem 4.2 (Section 4.2) that a single PFA can simultaneously witness $A_P(x)$ for every one of an infinite family of strings x of similar structure. This ability arguably lends A_P a descriptive advantage over a notion resulting from viewing a PFA as only describing its most probable string.

3. First results on A_P

In this section we establish a few basic properties of A_P and $A_{P,\delta}$, beginning by relating them to A_D and A_N:

Proposition 3.1. *(i) For any x, $A_P(x) \leq A_N(x) + 1$.*

(ii) $A_P(x) \leq A_{P,\delta}(x) \leq A_D(x)$ for all x and $\delta \in [0, 1)$. For every x, there is a $\delta' > 0$ such that $A_{P,\delta}(x) = A_P(x)$ for all $\delta \in [0, \delta')$.

Proof. (i) Let $M = (S, \Sigma, P, \vec{\pi}, \vec{\eta})$ be an NFA witnessing $A_N(x)$. Uniqueness of M's accepting path for x means in particular that $\vec{\pi}$ is a coordinate vector. Then define a PFA $M' = (S', \Sigma, P', \vec{\pi}', \vec{\eta}')$ as follows. Let $S' = S \cup \{s\}$, where s is a new state not occurring in S, to be listed after all other states. Let $\vec{\pi}' = [\vec{\pi}|0]$ and $\vec{\eta}' = [\vec{\eta}|0]$, where $[\vec{a}|\vec{b}]$ denotes the concatenation of the vectors $\vec{a}$ and $\vec{b}$. Write X^i for the ith row of any matrix X ($i \geq 1$). For each $\sigma \in \Sigma$, let P'_σ be built as follows from P_σ: if P^i_σ has at least one nonzero entry, let $(P'_\sigma)^i = [P^i_\sigma|0]/(\sum P^i_\sigma)$. Otherwise, let $(P'_\sigma)^i = [P^i_\sigma|1] = (0, \ldots, 0, 1)$. Finally, if $|S| = k$, then append a new row $(P'_\sigma)^{k+1} = (0, \ldots, 0, 1)$ (this corresponds to the new state s).

Then M' still has a unique accepting path of length $|x|$; in particular, x is the only string of length $|x|$ with $\rho_{M'}(x)$ positive. Therefore M' witnesses an upper bound for $A_P(x)$.

(ii) $\delta < \delta'$ implies $A_{P,\delta}(x) \leq A_{P,\delta'}(x)$, and if M is a DFA witnessing $A_D(x)$ then $\text{gap}_M(x) = 1$. This gives the first statement. For the second statement, one can for example pick $\delta' = \text{gap}_M(x)/2$ for any witness M for $A_P(x)$. □

Corollary 3.2. $A_P(x) \leq \lfloor |x| /2 \rfloor + 2$ *for all x.*

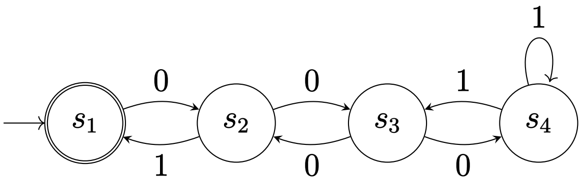


FIGURE 2. An NFA witnessing that $A_N(0001101) = 4$.

Proof. Hyde showed in [12, Theorem 3.1] that $A_N(x) \leq \lfloor |x|/2 \rfloor + 1$, so the bound immediately follows from the proposition. □

It follows from the procedure described in the first part of Proposition 3.1 that if $A_N(x)$ is witnessed by an NFA such that every state has at least one out-transition for every letter, then $A_P(x) \leq A_N(x)$ (because there are no rows of all zeros in the transition matrices, and the "dead state" s need not be added).

As an example of this construction, according to Bjørn Kjos-Hanssen's website,[2] $A_N(0001101) = 4$ via the NFA depicted in Figure 2. Here s_1 is both the initial and accepting state. In matrix form, this NFA can be represented as

$$P_0 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad P_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}, \quad \vec{\pi} = (1,0,0,0), \quad \vec{\eta} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

To transform this into a PFA, we need to add a fifth state due to the rows of zeros, and from the construction we get

$$P_0' = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1/2 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad P_1' = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1/2 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad \vec{\pi}' = (1,0,0,0,0), \quad \vec{\eta}' = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

There are strings for which the inequality in Proposition 3.1 is strict. Indeed, at the time of writing we know of no nonconstant strings such that $A_P(x)$ is equal to the maximum possible value $A_N(x)+1$. For example, $x = 0110$ has $A_N(x) = 3$, but $A_P(x) = 2$ by Theorem 4.1, as witnessed by the PFA with

$$\vec{\pi} = (1,0), \quad \vec{\eta} = (1,0)^T, \quad P_0 = \begin{pmatrix} 0 & 1 \\ 1/2 & 1/2 \end{pmatrix}, \quad P_1 = \begin{pmatrix} 1/2 & 1/2 \\ 0 & 1 \end{pmatrix}. \tag{1}$$

In fact, direct computations have shown that all binary strings x of length 9 or less have $A_P(x) \leq 3$, whereas many such strings have $A_N(x) = 4$ or 5. However, unlike with A_N, there are no strings x with $A_P(x) = 1$, because by the requirement for matrices to be row-stochastic and for the initial state distribution to be given by a probability vector, a PFA with one state must either accept all words with probability 1 or fail to accept any word. The only strings with $A_N(x) = 1$ are the constant strings $x = a^n$, and we have $A_P(a^n) = 2$ by Theorem 4.1.

So far we have not mentioned any examples involving $A_{P,\delta}$. Experimentally, it would appear that one has to make the value of δ quite low in order to get small values of $A_{P,\delta}$, for all but very short strings. This makes intuitive sense in view

[2] `https://math.hawaii.edu/wordpress/bjoern/complexity-of-0001101/`

of the proof of Theorem 4.2, and more generally the phenomenon of stability of a contractive iterated function system: all orbits converge to the attractor, and correspondingly acceptance probabilities will tend to cluster together for longer strings. As an example, if M is the PFA given in (1), then it follows from the proof of Theorem 4.2 that M witnesses $A_P(01^m0) = 2$ for all m, and one can calculate that $\mathrm{gap}_M(0110) = 1/16$, $\mathrm{gap}_M(01^30) = 1/32$, $\mathrm{gap}_M(01^40) = 1/64$, ...

It does not seem very easy to simultaneously make δ large and $A_{P,\delta}$ small even for short strings. One can get a gap of 7/16 for 0110 with the 3-state PFA

$$\vec{\pi} = (1,0,0), \quad \vec{\eta} = (0,0,1)^T, \quad P_0 = \begin{pmatrix} 0 & 0 & 1 \\ 1/4 & 3/4 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad P_1 = \begin{pmatrix} 0 & 3/4 & 1/4 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.$$

But among 2-state PFAs, the highest gap the author could find while still performing a feasible brute-force search was around 1/6. At the time of writing, the highest gap known for 0110 among 2-state PFAs is approximately 0.1775, via

$$\vec{\pi} = (1,0), \quad \vec{\eta} = (1,0), \quad P_0 = \begin{pmatrix} 0.16748 & 0.83252 \\ 0.99 & 0.01 \end{pmatrix}, \quad P_1 = \begin{pmatrix} 0.66116 & 0.33884 \\ 0 & 1 \end{pmatrix}.$$

This was found by numerical experimentation refining the result of a search of a set of roughly 850,000 2-state PFAs that turned up only one having gap greater than 1/6 (it was approximately 0.1719). Among the same set of PFAs, the largest gap found for 01^30 was approximately 0.1178.

The next result should be compared with the facts that $A_D(xyz) \geq A_D(y)$ and $A_N(xyz) \geq A_N(y)$ for any strings x, y, z (see [18, Lemma 12] and [12, Theorem 2.4]. The statement for A_N can be derived from $A_N(xy) \geq A_N(x)$ and the invariance of A_N under string reversal).

Proposition 3.3. *$A_{P,\delta}(xy) \geq A_{P,\delta}(y)$ for all $\delta \in [0,1)$ and strings x, y.*

Proof. Given δ, let $M = (S, \Sigma, P, \vec{\pi}, \vec{\eta})$ witness $A_{P,\delta}(xy)$. Let $M' = (S, \Sigma, P, \vec{\pi}', \vec{\eta})$ be a PFA with the same configuration as M except for its initial state distribution, which is now $\vec{\pi}' = \vec{\pi} P_M(x)$. Since $P_M(x)$ is a stochastic matrix, $\vec{\pi}'$ is still a probability vector. By definition, $P_M(xw) = P_M(x)P_M(w)$ for all strings w, so we have $\rho_{M'}(w) = \rho_M(xw)$ and consequently $\mathrm{gap}_{M'}(w) = \mathrm{gap}_M(xw)$ for all w. Hence M' witnesses an upper bound for $A_{P,\delta}(y)$: if not then there is a $w \neq y$, $|w| = |y|$, such that $\rho_{M'}(w) = \rho_M(xw)$ is at least $\rho_{M'}(y) = \rho_M(xy)$, a contradiction. □

One property of A_N which motivated its introduction, as mentioned above, is that $A_N(x) = A_N(\overleftarrow{x})$, where $\overleftarrow{x}$ is the reversal of x:

$$\overleftarrow{x} = x_n \cdots x_2 x_1 \quad \text{if} \quad x = x_1 x_2 \cdots x_n.$$

By Theorem 4.1, the class of strings x with $A_P(x) = 2$ is not closed under reversal, so A_P does not share this property. In Section 6 we will take up the question of how one might recover the property by modifying A_P. Equality of $A_P(x)$ and $A_P(\overleftarrow{x})$ is possible in at least some cases:

Proposition 3.4. *If $A_P(x)$ is witnessed by a PFA $M = (S, \Sigma, P, \vec{\pi}, \vec{\eta})$ such that each $P_\sigma \in P$ is doubly stochastic, and such that all nonzero entries of $\vec{\pi}$ are equal, then $A_P(\overleftarrow{x}) = A_P(x)$. If one can additionally take $\vec{\pi}$ and $\vec{\eta}$ to have the same number of nonzero entries, then $A_{P,\delta}(\overleftarrow{x}) = A_{P,\delta}(x)$ for all δ.*

Proof. The idea is more or less the content of Exercise A.2.8 of Chapter 3 of [20]. Define the PFA $M' = (S, \Sigma, P', \vec{\pi}', \vec{\eta}')$ by $P'_\sigma = P^T_\sigma$ for each $\sigma \in \Sigma$ and $\vec{\pi}' = \vec{\eta}^T/s$, where $s = \sum \vec{\eta}$. If each entry of $\vec{\pi}$ is either 0 or $1/n$ (for some $n \geq 1$), then let $\vec{\eta}' = n\vec{\pi}^T$.

M' intuitively represents the automaton obtained by operating M in reverse. We have $P_{M'}(\overleftarrow{x}) = P_M(x)^T$, so

$$\begin{aligned}\rho_{M'}(\overleftarrow{x}) = (\vec{\eta}^T/s)P_{M'}(\overleftarrow{x})(n\vec{\pi}^T) &= ns^{-1}\left(\vec{\pi}P_M(x)\vec{\eta}\right)^T \\ &= ns^{-1}\left(\rho_M(x)\right)^T = ns^{-1}\rho_M(x).\end{aligned}$$

The same calculation shows $\rho_{M'}(\overleftarrow{y}) = ns^{-1}\rho_M(y)$ for all y, so if $\rho_M(x) > \rho_M(y)$, then $\rho_{M'}(\overleftarrow{x}) > \rho_{M'}(\overleftarrow{y})$. Therefore M' witnesses $A_P(\overleftarrow{x}) \leq A_P(x)$ since M' and M have the same number of states. The opposite inequality follows by symmetry, so $A_P(\overleftarrow{x}) = A_P(x)$.

If $\vec{\pi}$ and $\vec{\eta}$ have the same number of nonzero entries, then $n = s$, and so $\rho_{M'}(\overleftarrow{y}) = \rho_M(y)$ for all $y \in \Sigma^*$. Hence $\mathrm{gap}_{M'}(\overleftarrow{x}) = \mathrm{gap}_M(x)$ and the second statement follows. □

As a corollary of this fact together with Theorem 4.1 below, for most binary strings x such that $A_P(x) = 2$, the latter cannot be witnessed by a PFA as in the proposition.

4. Classification of binary strings with $A_P = 2$

This section is devoted to proving the following theorem:

Theorem 4.1. *For a binary string w, $A_P(w) = 2$ if and only if w is of the form*

$$i^n j^m, \qquad i^n j^m i, \qquad i^n (ji)^m, \qquad \text{or} \quad i^n j(ij)^m$$

for some $n, m \geq 0$, where $i, j \in \{0, 1\}$.

This set of strings is significantly larger than the set of binary strings with NFA complexity 2. As classified in [12], the strings with $A_N(w) = 2$ consist exactly of

$$(ij)^m, \qquad i^m j, \qquad \text{and} \quad ij^m$$

for all m. All that can be generally said about $A_N(i^n j^m)$, for instance, is that it is no more than $\min\{n, m\} + 1$ [12, Example 4.1].

The proof of Theorem 4.1 will occupy a substantial portion of the rest of the paper, and we split it into two halves, the forward and reverse directions:

Theorem 4.2. *For a binary string w, if $A_P(w) = 2$, then w is of the form*

$$i^n j^m, \qquad i^n j^m i, \qquad i^n (ji)^m, \qquad \text{or} \quad i^n j(ij)^m \qquad \text{for some } n, m \geq 0.$$

Theorem 4.3. *$A_P(i^n j^m)$, $A_P(i^n j^m i)$, $A_P(i^n (ji)^m)$, and $A_P(i^n j(ij)^m)$ are equal to 2 for all $n, m \geq 0$.*

The proof depends on a connection between PFAs and iterated function systems, and we begin by giving the details of this connection in Section 4.1. Then we prove Theorem 4.2 in Section 4.2 and Theorem 4.3 in Section 4.3, ending by collecting some corollaries and further questions in Section 4.4.

4.1. **The iterated function system approach.** An *iterated function system* (IFS) on a compact metric space X is a dynamical system consisting of a finite set of continuous maps $f_1, \dots, f_n$ on X, viewed as inducing a semigroup action on X under composition. If X is $\mathbb{R}^n$ or a compact subset of it, and the maps f_i are affine maps, then the IFS is called *affine.* It is well-known that the attractors of many contractive IFSs are fractals, and the use of affine IFSs for efficient representation of fractal images has been studied [2, 7, 24].

Our interest in IFSs is, for present purposes, limited to the fact that one may obtain an IFS through the acceptance probability function of a PFA, and in doing so shed light on the family of strings whose complexity the PFA witnesses. Connections between IFSs and PFAs are already known: Culik and Dube [6, 7] in effect use PFAs as one method of generating fractal images, as an alternative to directly employing IFSs. They also introduce probabilistic affine automata, a generalization of PFAs in which each input letter corresponds to an affine map to be applied with some probability. (See [22] for a more recent study of this idea.)

Kocić and Simoncelli in [19] demonstrated a correspondence between IFSs given by a set of stochastic matrices and affine IFSs on lower-dimensional simplices. We present this correspondence in a more elementary formulation adapted to PFAs, showing that the PFA's acceptance probability function descends to the IFS in a natural fashion. Let $M = (S, \Sigma, P, \vec{\pi}, \vec{\eta})$ be a PFA with k states. If there are 0 or k accepting states, then ρ_M is identically 0 or 1, respectively, so assume without loss of generality that there are between 1 and $k-1$ accepting states. By permuting the states of M (and hence the rows and columns of $\vec{\pi}$, $\vec{\eta}$, and each P_σ), we may assume that the kth state is not accepting.

Recall that if $|w| = n$, then $\rho_M(w) = \vec{\pi} P_M(w) \vec{\eta}$, where $P_M(w) = \prod_{i=1}^n P_{w_{n-i}}$. This just means that $\rho_M(w)$ is a sum of up to $k-1$ elements of the row vector $\vec{\pi} P_M(w)$. We can think of each multiplication by a P_σ as updating the state distribution $\vec{\pi}$, and of $\vec{\pi}$ itself as representing the state distribution $\vec{\pi}(\lambda)$ after reading the empty string λ. Then let

$$\vec{\pi}(w) = \left(p_1(w), p_2(w), \dots, p_{k-1}(w), 1 - \sum_{i<k} p_i(w) \right) = \vec{\pi}(\lambda) P_M(w)$$

be the state distribution after reading a string w. Now, the last component of $\vec{\pi}(w^\frown\sigma)$ only depends on its first $k-1$ components together with the first $k-1$ columns of P_σ. Since the kth state of M is not accepting, $\rho_M(w^\frown\sigma)$ thus depends only on the first $k-1$ components of $\vec{\pi}(w)$, and if we only care about recovering ρ_M then we can drop the kth component from $\vec{\pi}(w)$ without losing any information.

So, let $\vec{a}_i$ be the ith row of P_σ truncated to its first $k-1$ entries, let $\vec{y}(w) = \vec{\pi}(w) \restriction (k-1)$, and let $\vec{1}_{m,n}$ be the $m \times n$ matrix of all 1s. Also let U be P_σ with its last and column deleted (so the rows of U are the vectors $\vec{a}_i$ for $i < k$). Then for any $\sigma \in \Sigma$,

$$\vec{y}(w^\frown\sigma) = \vec{y}(w) U + \left(1 - \sum \vec{y}(w)\right) \vec{a}_k = \vec{a}_k + \vec{y}(w) \left(U - \vec{1}_{k-1,1} \vec{a}_k\right).$$

$\vec{y}(w)$ is an element of the $(k-1)$-dimensional unit simplex S_{k-1}, so we identify $w \mapsto \vec{\pi}(w^\frown\sigma)$ with the map $f_\sigma \colon S_{k-1} \to S_{k-1}$ that sends $\vec{x}$ to $\vec{a}_k + \vec{x}B$, where $B = U - \vec{1}_{k-1,1}\vec{a}_k$. Note that the entries of B may be negative. Multiplication by P_σ thus corresponds to composition by f_σ. If we give the IFS consisting of the functions f_σ the starting vector $\vec{x}_0 = (p_1(\lambda), p_2(\lambda), \dots, p_{k-1}(\lambda))$, then we have

$$\rho_M(w) = \sum \Big\{ (f_{w_n} \circ f_{w_{n-1}} \circ \cdots \circ f_{w_0}(\vec{x}_0))_i : \text{the } i\text{th state of } M \text{ is accepting} \Big\}, \tag{2}$$

where $\vec{v}_i$ here denotes the ith component of the vector $\vec{v}$ and where $w = w_0 w_1 \cdots w_n$. Hence for any k-state PFA M there is an affine IFS on S_{k-1} whose iterations exactly recover the function ρ_M in the above fashion.

In the other direction, suppose we are given a finite set of affine maps $f_\sigma \colon \vec{x} \mapsto \vec{a} + \vec{x}B$ on S_{k-1}, where $\vec{a}$ and B depend on σ, along with a starting vector $\vec{x}_0 = (p_1, \dots, p_{k-1})$. We build a PFA M as follows. Let $\tilde{A}$ and $\tilde{B}$ be the $k \times k$ matrices given by $\tilde{A} = \vec{1}_{k,1} \left(\vec{a} \mid 1 - \sum \vec{a} \right)$ and

$$\tilde{B} = \begin{pmatrix} \vec{1}_{k-1,1} \\ 0 \end{pmatrix} \left(B \mid -B\vec{1}_{k-1,1} \right) = \left(\begin{array}{c|c} & -\sum_{i<k} B_{1,i} \\ B & \vdots \\ & -\sum_{i<k} B_{k-1,i} \\ \hline \vec{0} & 0 \end{array} \right).$$

Then let

$$P_\sigma = \tilde{A} + \tilde{B} \quad \text{and} \quad \vec{\pi} = \vec{\pi}(\lambda) = \left(\vec{x}_0 \mid 1 - \sum \vec{x}_0 \right) \in \mathbb{R}^k.$$

Also define $\vec{\pi}(w)$ for any w as before. Then P_σ is stochastic: first, each row clearly sums to 1 as the row sums of $\tilde{A}$ and $\tilde{B}$ are all 1 and 0, respectively. If $\vec{e}_i$ is the ith standard basis vector in $\mathbb{R}^{k-1}$, then $f_\sigma(\vec{e}_i)$ is the sum of $\vec{a}$ and the ith row of B, i.e., the upper left $(k-1) \times (k-1)$ submatrix of P_σ. From $\vec{a} = f_\sigma(\vec{0}) \in S_{k-1}$ and $f_\sigma(\vec{e}_i) \in S_{k-1}$ it follows that each entry of $\tilde{A} + \tilde{B}$ is in $[0,1]$.

One can check that $\vec{\pi}(\lambda) P_\sigma \upharpoonright (k-1) = \vec{a} + \vec{x}_0 B = f_\sigma(\vec{x}_0)$. Inductively we have that $\vec{\pi}(w^\frown\sigma) \upharpoonright (k-1) = f_\sigma(\vec{x})$ if $\vec{x} = \vec{\pi}(\lambda) P_M(w)$. Now, the data we have so far does not uniquely specify a PFA $M = (\{1, \dots, k\}, \Sigma, \{P_\sigma\}, \vec{\pi}, \vec{\eta})$, because nothing about the vector of accepting states $\vec{\eta}$ is implied by the IFS we started with except that the kth state should not be accepting. Thus the same IFS can be made to correspond to any PFA M having the $\vec{\pi}$ and matrices P_σ given above, and such that the last state is not accepting. The equation (2) holds for any such M and w, which completes the correspondence.

Since we will only apply this correspondence to two-state automata in the present work, we separately outline this case for clarity. Given a two-state PFA M, write $\vec{\pi}$ as $(p, 1-p)$. Assume by permuting the states that $\vec{\eta} = (1, 0)^T$. For each $\sigma \in \Sigma$, write

$$P_\sigma = \begin{pmatrix} a_\sigma + b_\sigma & 1 - a_\sigma - b_\sigma \\ a_\sigma & 1 - a_\sigma \end{pmatrix}, \tag{3}$$

where b_σ is allowed to be negative. Then for each $w \in \Sigma^*$, we have

$$\rho(w^\frown\sigma) = \begin{pmatrix} \rho(w) & 1 - \rho(w) \end{pmatrix} \begin{pmatrix} a_\sigma + b_\sigma & 1 - a_\sigma - b_\sigma \\ a_\sigma & 1 - a_\sigma \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = a_\sigma + b_\sigma \rho(w).$$

We can thus associate to P_σ the "incremental probability function"

$$f_\sigma(x) = a_\sigma + b_\sigma x$$

mapping $[0,1]$ into itself. Viewing p as $\rho(\lambda)$, we obtain the IFS $(f_\sigma)_{\sigma \in \Sigma}$ with starting value $x_0 = p$ such that for any word $w = w_1 w_2 \dots w_n$,

$$\rho(w) = f_{w_n} \circ f_{w_{n-1}} \circ \cdots \circ f_{w_1}(x_0). \tag{4}$$

In the other direction, starting from an IFS given by affine maps f_σ on $[0,1]$ together with x_0, setting $\vec{\pi} = (x_0, 1-x_0)$ and defining the matrices P_σ as in (3) produces a PFA whose acceptance probability function satisfies (4).

The set of w such that an upper bound for $A_P(w)$ is witnessed by M is exactly the set of w describing a sequence of compositions maximizing the value along the orbit of x_0 under this IFS. This idea will be exploited heavily throughout the following section.

4.2. **Proof of Theorem** 4.2. We will establish that any two-state PFA over a binary alphabet must witness the complexity of only strings in one of the forms given in the theorem, i.e.,

$$i^n j^m, \qquad i^n j^m i, \qquad i^n (ji)^m, \qquad \text{or} \quad i^n j(ij)^m,$$

if it witnesses anything at all. If $i = j$, of course, these strings are constant and so trivially have complexity 2. Permuting the underlying alphabet does not change the complexity of a string, as it corresponds merely to permuting the maps f_σ of the IFS (or equivalently the transition matrices P_σ of the original PFA). Therefore, any statement in this section about a string should be understood to apply equally well to its bit-flip (i.e., the result of permuting 0 and 1), by switching the roles of f_0 and f_1.

Assume we are given a two-state PFA represented by the IFS

$$f_0(x) = a + bx \qquad \text{and} \qquad f_1(x) = c + dx$$

with starting value $x_0 \in [0,1]$, where f_0 and f_1 map $[0,1]$ into itself. We use the word "orbit" to mean any forward orbit of x_0 under the semigroup action generated by f_0 and f_1, that is, the orbit of x_0 under some particular sequence of compositions of f_0 and f_1. We always omit parentheses when composing functions, so that e.g. $f_0^2 x = f_0(f_0(x))$. For brevity, we describe a probability as *n-maximal* (*n-minimal*) if it is maximal (minimal) among the probabilities of strings of length n. We also refer to an n-maximal (n-minimal) probability as simply an n-maximum (n-minimum). If n is clear from context, we may call such a probability maximal (minimal) or a maximum (minimum). We say the IFS witnesses a string w if it witnesses that $A_P(w) = 2$, i.e., $\rho(w)$ is maximal. The following basic observations will be useful throughout:

- For $i \in \{0,1\}$, if f_i is not the line $y = x$, then f_i has a unique fixed point in $[0,1]$, towards which it contracts with rate equal to the absolute value of its slope. The case $y = x$ will be dispensed with in Lemma 4.4, and we can assume elsewhere that neither f_0 nor f_1 is the identity map. We will use r_0 and r_1 to denote the fixed points of f_0 and f_1, respectively. By abuse of notation, r_0 and r_1 refer either to the x-coordinates of these points or to the actual points in $[0,1]^2$. It will be clear which is meant from the context. We have $r_0 = a/(1-b)$ and $r_1 = c/(1-d)$.
- If f_i has positive slope, then it maps $[0, r_i)$ into itself and $(r_i, 1]$ into itself. If it has negative slope, it maps $[0, r_i)$ into $(r_i, 1]$ and vice versa. (If its slope is 0, of course, it sends every point to r_i.)
- If a probability x is n-maximal, then x is either the image of an $(n-1)$-maximum under a map of positive slope, or the image of an $(n-1)$-minimum under a map of negative slope. Hence we need only consider the maximum

and minimum probabilities of each length in order to determine the maximal-probability strings.

- Suppose f_0 and f_1 intersect at the single point $(i_x, i_y) \in [0,1]^2$, and that the maximum or minimum probability of some length turns out to equal i_x. (We always assume the maps do not coincide, since no strings can be witnessed if they do.) Then no further probabilities in the same orbit can be unique (since $f_0 i_x = f_1 i_x$). In this case, no further strings are witnessed if their probabilities are in the same orbit as i_x. We assume for simplicity that this does not happen in the arguments that follow. This does not lose any generality, because if i_x happens to be attained as the maximal or minimal probability in some orbit, then nothing changes about the behavior of the IFS except for the lack of uniqueness of the subsequent maxima and minima.

For clarity, we separate the argument into several progressively more complicated cases based on the signs of b and d. First we quickly dispense with some easy ones: if both f_0 and f_1 are constant, the PFA witnesses either 0^n or 1^n for all n, depending on which line is higher. If one map is the identity, or more generally if f_0 and f_1 commute, no strings can be witnessed beyond constant strings, because the only determining factor of $\rho(w)$ is the number of 0s and 1s in w.

The following lemma collects various observations which will be useful for the rest of the proof.

Lemma 4.4. *Let $f_0 = a + bx$ and $f_1 = c + dx$ be maps from the unit interval into itself. Assume by convention that $a < c$ and that the maps intersect at the unique point $(i_x, i_y) \in [0,1]^2$. Taken together, these imply in particular that $b > d$, and we will always assume $a < c$ and $b > d$.*

(a) If neither map is the identity, then either both maps fix i_x or neither does. We always assume neither map is the identity from this point on, as well as that the maps do not coincide.

(b) Both maps fix i_x, i.e., $i_y = i_x$, iff $r_0 = r_1 = i_x$ iff $f_0 f_1 = f_1 f_0$.

(c) Both maps decrease i_x, i.e., $i_y < i_x$, iff $r_0 < r_1 < i_x$ iff $f_0 f_1 < f_1 f_0$ iff $f_0 f_1^2 < f_1^2 f_0$.

(d) Both maps increase i_x, i.e., $i_y > i_x$, iff $r_0 > r_1 > i_x$ iff $f_1 f_0 < f_0 f_1$ iff $f_1^2 f_0 < f_0 f_1^2$.

(e) If f_0 and f_1 both have negative slopes, and if neither fixes i_x, then $|r_0 - r_1| < |r_1 - i_x|$.

(f) Suppose f_0 and f_1 both have negative slopes. If both maps decrease i_x, then if $x \in [r_0, i_x)$, every orbit of x always stays inside $[0, i_x)$. If both maps increase i_x, then if $x \in (i_x, r_0]$, every orbit of x always stays inside $(i_x, 1]$.

Proof. (a) r_0 and r_1 are the intersections of f_0 and f_1, respectively, with $y = x$. If (say) $r_0 = i_x$, then (i_x, i_x) also lies on $y = x$ and is in the range of f_1, therefore $r_1 = i_x$ too.

(b) The first equivalence is obvious by definition. Notice $f_0 f_1 x = a + bc + bdx$ and $f_1 f_0 x = c + ad + bdx$. Then $f_0 f_1 = f_1 f_0 \iff a + bc = c + ad \iff a/(1-b) = c/(1-d)$, i.e., iff $r_0 = r_1$, and this happens iff they both equal i_x since r_0 and r_1 both lie on the line $y = x$.

(c) Both r_0 and r_1 are less than i_x in this case, because the maps contract towards their fixed points, so if $f_i x < x$ then $x > r_i$. We have $i_x = (c-a)/(b-d)$ and $i_y = (bc - ad)/(b-d)$. Remembering that our assumptions imply $b > d$ no matter the sign of each, and observing that neither b nor d equals 1, we have $i_x > i_y$ iff

$$\frac{c-a}{b-d} > \frac{bc-ad}{b-d} \iff c-a > bc-ad \iff c(1-b) > a(1-d)$$
$$\iff \frac{c}{1-d} > \frac{a}{1-b} \iff r_1 > r_0.$$

Since $r_1 > r_0 \iff c+ad > a+bc \iff f_1f_0 > f_0f_1$, this completes the second equivalence. For the third, note $f_0f_1^2x = a + b(c+cd+d^2x)$ and $f_1^2f_0x = c+cd+d^2(a+bx)$. Then

$$f_0f_1^2x < f_1^2f_0x \iff a+bc+bcd+bd^2x < c+cd+ad^2+bd^2x$$
$$\iff d(bc-ad) < (c-a)-c(b-d) \iff c+d\frac{bc-ad}{b-d} < \frac{c-a}{b-d}$$
$$\iff c+di_y < i_x \iff f_1i_y < i_x \iff f_1^2i_x < i_x.$$

The last inequality holds iff $r_1 < i_x$ (as f_1^2 contracts i_x towards its fixed point r_1), which happens iff $i_x > i_y$ by the first equivalence.

(d) This follows by swapping "$>$" with "$<$" everywhere in the argument for part (c).

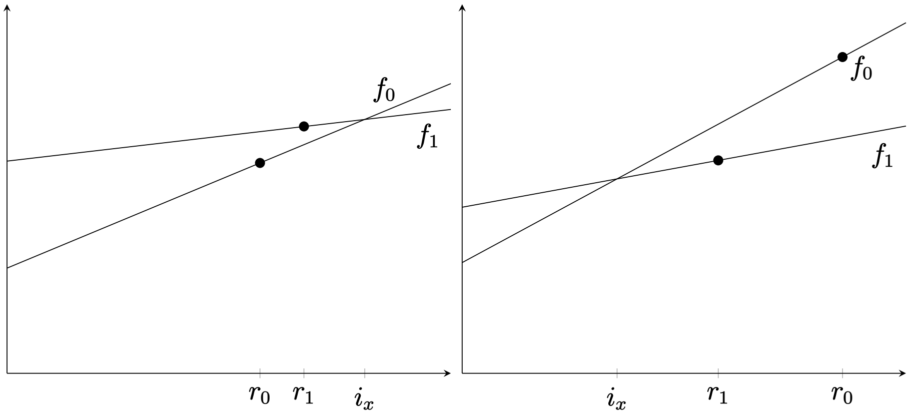
(e) By writing $c = r_1(1-d)$, one can rearrange the formula $i_y = c+di_x$ to obtain $d = (i_y-r_1)/(i_x-r_1)$. Since $|d| \leq 1$, this implies $|i_y - r_1| \leq |i_x - r_1|$. We will finish the proof by showing that when $b < 0$, then in fact $i_x > i_y$ iff $i_y < r_0$ and $i_x < i_y$ iff $i_y > r_0$. That is, depending on whether both maps decrease or increase i_x, we have either $i_y < r_0 < r_1 < i_x$ or $i_y > r_0 > r_1 > i_x$. Then $|i_y - r_1| = |i_y - r_0| + |r_0 - r_1| > |r_0 - r_1|$, and one gets $|r_0 - r_1| < |i_y - r_1| \leq |i_x - r_1|$.

So, $i_y < r_0$ if and only if

$$\frac{bc-ad}{b-d} < \frac{a}{1-b} \iff (bc-ad)(1-b) < a(b-d) \iff bc-b^2c+abd < ab$$
$$\iff bc(1-b) < ab(1-d) \iff c(1-b) > a(1-d) \iff \frac{c}{1-d} > \frac{a}{1-b},$$

i.e., if and only if $r_1 > r_0$, which is equivalent to $i_x > i_y$. (The change from $<$ to $>$ in the second line is because $b < 0$.) It is clear that one can switch "$<$" and "$>$" everywhere in this argument to obtain that $i_y > r_0$ iff $i_x < i_y$, and the proof is complete.

(f) For the first claim, by assumption $i_y < i_x$ and so $r_0 < r_1 < i_x$. Since $f_0i_x = f_1i_x$, we have $f_0f_1i_x = f_0f_0i_x$, which is less than i_x. This implies that r_{01}, the fixed point of f_0f_1, is also less than i_x: if f_0f_1 decreases the value of a point, then that point must be above r_{01}. As f_0f_1 contracts to r_{01}, we have that $f_0f_1x < i_x$ whenever $x < i_x$. In other words, if $\rho(w) < i_x$, then $\rho(w^\frown 10) < i_x$ too. The analogous statement holds for f_1f_0, i.e., $\rho(w) < i_x$ implies $\rho(w^\frown 01) < i_x$. Finally, since $|r_0 - r_1| < |r_1 - i_x|$ by part (e), f_1 always sends points in $[r_0, i_x)$ to points below i_x (and of course the same statement is clearly true for f_0). This is clear if $x \in [r_1, i_x)$. If $x \in [r_0, r_1)$, then $|f_1x - r_1| < |x - r_1| < |r_0 - r_1| < |i_x - r_1|$, so f_1x is closer to r_1 than i_x is, and must be less than i_x. Overall, then, we have that once an orbit enters $[r_0, i_x)$, it stays below i_x. The second claim can be proven by switching "$<$" and "$>$" everywhere in the above argument. □



(A) Both maps decrease i_x (B) Both maps increase i_x

FIGURE 3. Subcases for f_0, f_1 with positive slope (Case 2)

Now begins the main body of the proof of Theorem 4.2. It is split into four cases: the maps do not intersect, they intersect and have positive slope, they intersect and have negative slope, and they intersect with one having positive and the other having negative slope. The last three cases are each split into two further subcases, based on whether the maps increase or decrease i_x.

Case 1: f_0 and f_1 do not intersect in $[0,1]$. Suppose without loss of generality that $f_1 x > f_0 x$ for all $x \in [0,1]$, so that $a < c$. If both maps have positive slope, it follows that $\rho(1^n)$ is maximal for all n. If both have negative slope, then appending 0 to a maximal probability always leads to a minimal probability, and appending 1 to a minimal probability always leads to a maximal probability. Therefore $(01)^n$ and $1(01)^n$ are witnessed for all n, as $f_0 x_0 < f_1 x_0$. If f_1 has positive and f_0 has negative slope, 1^n is witnessed for all n; note the ranges of f_0 and f_1 cannot overlap here. And if f_1 has negative and f_0 positive slope, then $0^n 1$ is witnessed for all n: a minimum can only be reached by adding all 0s, again because the ranges are disjoint, and since $f_0 f_1 x < f_1 f_0 x$ for all x (Lemma 4.4(c)), appending 10 to a string always gives a lower probability than appending 01 does.

Strings witnessed in this case: 1^n, $(01)^n$, $1(01)^n$, $0^n 1$.

From now on, assume that $a < c$, that f_0 and f_1 intersect at the point $(i_x, i_y) \in [0,1]^2$, and that the maps do not commute. By Lemma 4.4, this implies that r_0, r_1, and i_x are distinct. Assuming $a < c$ is no loss of generality, because we are working only up to permuting f_0 and f_1, and $a < c$ is required for the maps to intersect within the unit interval. As noted in the lemma, this also implies $b > d$ in every case.

Case 2: f_0 and f_1 both have nonnegative slope. Specifically, assume $a, b, c \geq 0$ and $d > 0$ (the case $b = d = 0$ is trivial). There are then two possible subcases of this case, which are illustrated in Figure 3:

(a) Both f_0 and f_1 decrease i_x. Then we must have $r_0 < r_1 < i_x$, and $0^{n_0} 1^m$ is witnessed for all m, where $n_0 \geq 0$ is least such that $f_0^{n_0} x_0 < i_x$ (taking f_0^0 to be the identity map). This is because $f_0 x > f_1 x$ for $x > i_x$, but iterating it will eventually cause the value to drop below i_x, and from that point on, $f_1 > f_0$. If $x_0 < i_x$ then we have $f_1 > f_0$ from the start.

(b) Both f_0 and f_1 increase i_x. Then $i_x < r_1 < r_0$, and $1^{n_0}0^m$ is witnessed for all m, where $n_0 \geq 0$ is least such that $f_1^{n_0}x_0 > i_x$. The reasoning is exactly analogous to that in (a).

Strings witnessed in the above case: 1^n, 0^n1^m, 1^n0^m.

Case 3: f_0 and f_1 both have negative slope. The special case $b = 0$ and $d < 0$ is discussed under Case 4 below, so assume b and d are both strictly negative here. Recall that having negative slopes means each f_i "flips x over" r_i: if $x < r_i$, then $f_ix > r_i$, and vice versa. If we start an orbit with a maximal probability of its length, then the orbit can only lead to maximal probabilities for odd-length strings (and this is the only way to witness odd-length strings). This is accomplished by extending a string on even lengths in order to achieve a *minimal* probability. For the same reason, if we start an orbit with a minimal probability, then only even-length strings may have maximal probabilities in that orbit. The two essentially different cases for the configuration of f_0 and f_1 are shown in Figure 4.

(a) Both f_0 and f_1 decrease i_x. Then $r_0 < r_1 < i_x$. Recall that by Lemma 4.4(f), once an orbit enters $[r_0, i_x)$, it stays below i_x forever.

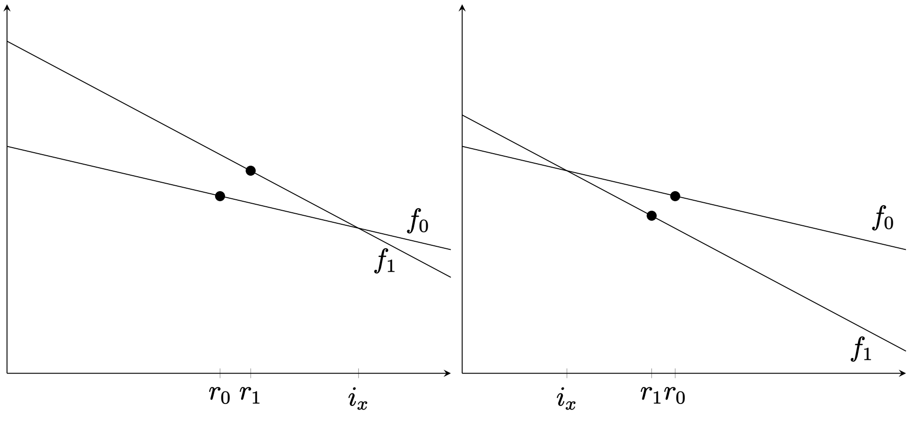
Suppose $x_0 > i_x$ and we start an orbit with the maximal-probability string 0. Then $\rho(0) < i_x$, and $\rho(0^2)$ is minimal. If $\rho(0^2) > i_x$, then $\rho(0^3)$ is maximal, since $f_0 > f_1$ above i_x. Every $(2\ell + 1)$-maximal probability is an image of a 2ℓ-minimal probability, so as long as $\rho(0^{2\ell}) > i_x$, we have that $\rho(0^{2\ell+1})$ is maximal and $\rho(0^{2\ell+2})$ is minimal. Let n_0 be least such that $\rho(0^{2n_0}) < i_x$. Once this happens, since $\rho(0^{2n_0})$ is minimal and $f_0 < f_1$ below i_x, $\rho(0^{2n_0}1)$ is maximal.

$\rho(0^{2n_0})$ is between r_0 and i_x, so we know that its future orbit will always stay below i_x by Lemma 4.4(f). This means that a maximum is always reached by appending 1 to a minimum, and a minimum is always reached by appending 0 to a maximum. Hence, among odd-length strings with length greater than $2n_0 + 1$, we witness $0^{2n_0}1(01)^m$ for all $m \geq 0$.

Next, say $x_0 > i_x$ and we start our orbit with the minimal-probability string 1. Then $\rho(11)$ is maximal. If $\rho(11) > i_x$, then $\rho(1^3)$ is minimal and $\rho(1^4)$ is maximal. So we initially witness $1^{2\ell}$ for $\ell \leq n_0$, where n_0 is least such that $\rho(1^{2n_0}) < i_x$. Among longer strings, we then witness $1^{2n_0}(01)^m$ for all m. To see this, one argues in a similar way as when $x_0 < i_x$, since $f_1f_0x < i_x$ when $x \in [r_0, i_x)$. The only difference is that once $\rho(1^{2n_0}) < i_x$, the $(2n_0 + 1)$-minimal probability is attained by $\rho(1^{2n_0}0)$, as $f_0x < f_1x$ for $x < i_x$. Then appending a 1 gives the $(2n_0 + 2)$-maximum $\rho(1^{2n_0}01)$, and we continue appending 01 to keep the min-max pattern going and get $(2n_0 + 2k)$-maximal probabilities for all k. This concludes the subcase $x_0 > i_x$.

Finally, suppose $x_0 < i_x$. This is analogous to the case $x_0 > i_x$, but with even-odd parity swapped everywhere. In fact, we only need consider the case $x_0 < r_0$, because when $x_0 \in [r_0, i_x)$, we know that all orbits stay below i_x, so for such x_0 we witness $(10)^m$ and $0(10)^m$ for all $m \geq 0$.

Now, if $x_0 < r_0$ and we start with the maximal-probability string 1, we at first witness $1^{2\ell+1}$ among odd-length strings, as long as $\rho(1^{2\ell+1}) > i_x$. If n_0 is least such that $\rho(1^{2n_0+1}) < i_x$, then we witness 1^{2n_0+1} and subsequently $1^{2n_0+1}(01)^m$ for all $m \geq 1$. This is because once $\rho(1^{2n_0+1}) < i_x$, then the $(2n_0 + 2)$-minimum is $\rho(1^{2n_0+1}0)$, followed by the $(2n_0 + 3)$-maximum $\rho(1^{2n_0+1}01)$, and continuing to append 01 keeps the min-max pattern going. Starting instead with the minimal-probability 0, we witness $0^{2\ell+2}$ among even-length strings as long as



(A) Both maps decrease i_x (B) Both maps increase i_x

FIGURE 4. Subcases for f_0, f_1 with negative slope (Case 3)

$\rho(0^{2\ell+1}) > i_x$. If n_0 is least such that $\rho(0^{2n_0+1}) < i_x$, then $\rho(0^{2n_0+1})$ is minimal but $\rho(0^{2n_0+2})$ is not maximal. Therefore $\rho(0^{2n_0+1}1)$ must be maximal, and we witness $0^{2n_0+1}1(01)^m$ for all $m \geq 0$. The pattern of appending 01 can be repeated forever to obtain maximal probabilities because $\rho(0^{2n_0+1}1) \in [r_0, i_x)$ and applying $f_1 f_0$ will always stay below i_x, where f_0 is minimal and f_1 is maximal.

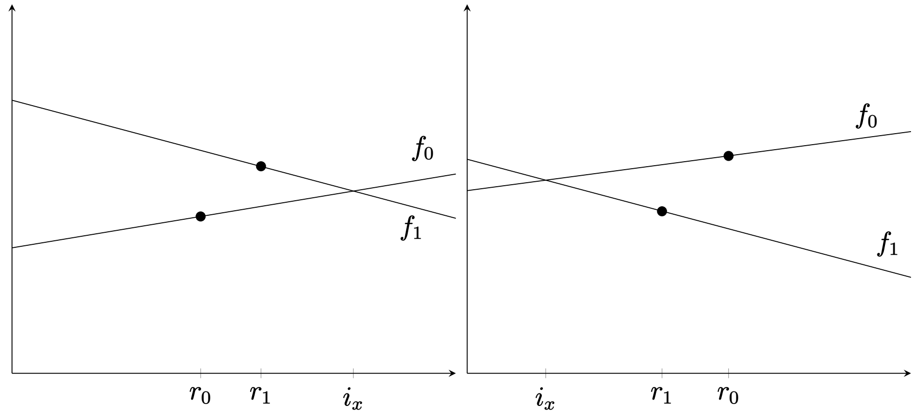
Strings witnessed in the above case: 0^{2m}, 1^{2m+1}, $0^{2n}1(01)^m$, $1^{2n}(01)^m$, $0^{2n+1}1(01)^m$.

(b) Both f_0 and f_1 increase i_x. Then $i_x < r_1 < r_0$. By Lemma 4.4(f), once an orbit enters $(i_x, r_0]$, it stays above i_x.

Let $x_0 < i_x$. By starting with the maximal $\rho(1)$, we have that $\rho(1^{2\ell+1})$ is maximal as long as $\rho(1^{2\ell}) < i_x$. (Note that $\rho(1^{2\ell+1})$ is always greater than r_1 and hence also i_x.) If n_0 is least such that $\rho(1^{2n_0}) > i_x$, then $\rho(1^{2n_0})$ is $2n_0$-minimal but $\rho(1^{2n_0+1})$ is not $(2n_0+1)$-maximal. Therefore $\rho(1^{2n_0}0)$ is $(2n_0+1)$-maximal, and since $\rho(1^{2n_0}) \in (i_x, r_0]$, we have that $\rho(1^{2n_0}0(10)^m)$ remains above i_x for all $m \geq 0$ and is therefore maximal. By starting instead with the minimal $\rho(0)$, we have that $\rho(0^{2\ell+2})$ is maximal as long as $\rho(0^{2\ell}) < i_x$. If n_0 is least such that $\rho(0^{2n_0}) > i_x$, then $\rho(0^{2n_0})$ is $2n_0$-maximal but $\rho(0^{2n_0+1})$ is not $(2n_0+1)$-minimal, since $f_0 > f_1$ for $x > i_x$. Therefore $\rho(0^{2n_0}1)$ is minimal, and because $\rho(0^{2n_0}) \in (i_x, r_0]$, its future orbits stay above i_x and we have $\rho(0^{2n_0}(10)^m)$ maximal for all $m \geq 0$.

If $x_0 \in (i_x, r_0]$, then we witness $(10)^m$ and $0(10)^m$ for all $m \geq 0$, as in case (a) when $x_0 \in [r_0, i_x)$. If $x_0 > r_0$ and we start with the maximal $\rho(0)$, then $\rho(0^{2\ell+3})$ is maximal and $\rho(0^{2\ell+2})$ is minimal as long as $\rho(0^{2\ell+1}) < i_x$. If n_0 is least such that $\rho(0^{2n_0+1}) > i_x$, then $\rho(0^{2n_0+1})$ is $(2n_0+1)$-maximal but $\rho(0^{2n_0+2})$ is not $(2n_0+2)$-minimal. Instead, $\rho(0^{2n_0+1}1)$ is $(2n_0+2)$-minimal, and since $\rho(0^{2n_0+1}) \in (i_x, r_0]$, all future orbits stay above i_x, where $f_0 > f_1$. Therefore $\rho(0^{2n_0+1}(10)^m)$ is maximal for all $m \geq 0$. If instead we start with the minimal $\rho(1)$, then $\rho(1^{2\ell+2})$ is maximal as long as $\rho(1^{2\ell+1}) < i_x$. If n_0 is least such that $\rho(1^{2n_0+1}) > i_x$, then $\rho(1^{2n_0+1})$ is $(2n_0+1)$-minimal but $\rho(1^{2n_0+2}) < \rho(1^{2n_0+1}0)$, which is now $(2n_0+2)$-maximal. Since $\rho(1^{2n_0+1}) \in (i_x, r_0]$, all of its future orbits stay above i_x, and thus $\rho(1^{2n_0+1}0(10)^m)$ is maximal for all $m \geq 0$.

Strings witnessed in the above case: 1^{2m+1}, $1^{2n}0(10)^m$, $0^{2n+1}(10)^m$, $1^{2n+1}0(10)^m$.



(A) Both maps decrease i_x (B) Both maps increase i_x

FIGURE 5. Subcases for f_0 with positive and f_1 with negative slope (Case 4)

Case 4: f_0 has positive slope and f_1 has negative slope. The basic possibilities are illustrated in Figure 5.

(a) Both f_0 and f_1 decrease i_x. Lemma 4.4 implies that this is equivalent to $f_1 f_0 x > f_0 f_1 x$ for all x, so that appending 01 always gives a higher probability than appending 10 would to the same string. Assume for now that $b > 0$; we will treat the special case $b = 0$ below. The general pattern when $x_0 > i_x$ will follow from the next three claims:

Claim 1. There is an n_0 such that $\rho(1^{2n_0+1}) \geq \rho(1^{2n_0-1}0^2)$.

Proof. f_1 contracts to r_1, which is greater than r_0. Therefore $\rho(1^{2n+1}) \geq r_0$ for some n. If that is the case, then either $\rho(1^{2n-1}) \leq r_0$ and so is $\rho(1^{2n-1}0^2)$, or if $\rho(1^{2n-1}) \geq r_0$, then appending 0^2 to 1^{2n-1} decreases the probability towards r_0 while appending 1^2 increases it towards r_1. □

From now on, take n_0 to be the least value as in the previous claim. If $x_0 > i_x$, then 0 is 1-maximal, and it follows that $n_0 \geq 1$. If $\rho(1^{2n_0+1}) = \rho(1^{2n_0-1}0^2)$, then both are minimal, so no further strings will be witnessed as there are no longer unique minima or maxima of any greater length. Hence without loss of generality assume the inequality is strict. The second and third claims will apply to the case $n_0 > 1$; the case $n_0 = 1$ is handled separately afterwards.

Claim 2. If $n_0 > 1$, then for all $1 \leq \ell \leq n_0$, we have:
(α) $\rho(1^{2\ell})$ is 2ℓ-maximal,
(β) $\rho(1^{2\ell-1}0)$ is 2ℓ-minimal,
(γ) $\rho(1^{2\ell-1}01)$ is $(2\ell+1)$-maximal, and
(δ) if $\ell < n_0$, then $\rho(1^{2\ell+1})$ is $(2\ell+1)$-minimal.

Proof. By induction on ℓ. For $\ell = 1$, because $\rho(0) > \rho(1)$, the only possible 2-maxima are $\rho(00)$ and $\rho(11)$. If we have $\rho(11) < \rho(00)$, then because $f_1 x < f_1 y$ iff $x > y$ for any x, y, also $f_1^3 x_0 > f_1 f_0^2 x_0$. But from $f_0 f_1 < f_1 f_0$ it follows that $f_1 f_0^2 x_0 > f_0 f_1 f_0 x_0 > f_0^2 f_1 x_0 = \rho(10^2)$, as $f_0 x < f_0 y$ iff $x < y$ for any x, y. (The latter is due to f_0 having positive slope.) Therefore $\rho(1^3) > \rho(10^2)$, or in other words $n_0 = 1$. Since we are assuming $n_0 > 1$, this is a contradiction, hence

$\rho(00) < \rho(11)$ and the latter is 2-maximal, which establishes the base case of (α).

Next, because $f_1 f_0 > f_0 f_1$, we have $\rho(01) > \rho(10)$. The latter is less than $\rho(00)$: $\rho(1) < \rho(0)$, so if $\rho(1) < r_0$ then $\rho(10) < r_0 < \rho(00)$. If $\rho(1) \geq r_0$, then appending a 0 moves $\rho(10)$ closer to r_0 than $\rho(00)$ is, i.e., makes it smaller than $\rho(00)$. This implies (β) holds for $\ell = 1$. Then (γ) follows if (α) and (β) hold for any ℓ: the only possible candidates for a $(2\ell+1)$-maximum are $\rho(1^{2\ell}0)$ and $\rho(1^{2\ell-1}01)$, i.e., the image of the 2ℓ-maximum under f_0 and the image of the 2ℓ-minimum under f_1. But $\rho(1^{2\ell}0) < \rho(1^{2\ell-1}01)$ since $f_0 f_1 < f_1 f_0$. For (δ), suppose (α) and (β) are true for ℓ and $\ell < n_0$. Then only $\rho(1^{2\ell-1}0^2)$ or $\rho(1^{2\ell+1})$ could possibly be minima, since they are the images of the 2ℓ-minimum under f_0 and the 2ℓ-maximum under f_1, respectively. And we have $\rho(1^{2\ell+1}) < \rho(1^{2\ell-1}0^2)$ because $\ell < n_0$.

Now suppose all four items hold for some given $\ell < n_0$. Then if (α) and (β) hold for $\ell + 1$, so does (γ) by the above argument, and if $\ell + 1 < n_0$ then additionally (δ) holds for $\ell + 1$. Hence, for the inductive step, it only remains to establish that (α) and (β) hold for $\ell + 1$. For (α), because a $(2\ell+2)$-maximum is either the image under f_1 of a $(2\ell+1)$-minimum or the image under f_0 of a $(2\ell+1)$-maximum, the only possible $(2\ell+2)$-maxima are $\rho(1^{2\ell+1}1)$ and $\rho(1^{2\ell-1}010)$. But we have $\rho(1^{2\ell-1}010) < \rho(1^{2\ell-1}001)$ because $f_0 f_1 < f_1 f_0$, so $\rho(1^{2\ell-1}010)$ is not maximal. Finally, for (β), $\rho(1^{2\ell+1}0)$ is $(2\ell+2)$-minimal because the only other possible candidate for a minimum is $\rho(1^{2\ell-1}011)$, and $\rho(1^{2\ell+1}0)$ is less than $\rho(1^{2\ell-1}011)$. The latter follows by Lemma 4.4, as $i_x > i_y$ if and only if $f_0 f_1^2 < f_1^2 f_0$, so that appending 110 always results in a lower probability than appending 011 to the same string. □

Claim 3. If $n_0 > 1$, then $\rho(1^{2n_0-1}0^m)$ is $(2n_0 - 1 + m)$-minimal, and hence $\rho(1^{2n_0-1}0^m1)$ is $(2n_0 + m)$-maximal, for all $m \geq 0$.

Proof. The cases $m = 0$ and $m = 1$ are covered by taking $\ell = n_0 - 1$ and $\ell = n_0$ in the previous claim. If $\rho(1^{2n_0-1}0^m)$ is $(2n_0 - 1 + m)$-minimal, and $\rho(1^{2n_0-1}0^{m-1}1)$ is $(2n_0 - 1 + m)$-maximal, then only $\rho(1^{2n_0-1}0^m0)$ or $\rho(1^{2n_0-1}0^{m-1}11)$ could be $(2n_0+m)$-minimal. But $f_0 f_1^2 < f_1^2 f_0$ implies $\rho(1^{2n_0-1}0^{m-2}110) < \rho(1^{2n_0-1}0^{m-1}11)$, so the latter is not minimal. Finally, this implies $\rho(1^{2n_0-1}0^{m+1}1)$ is $(2n_0+m+1)$-maximal: the only other possibility is $\rho(1^{2n_0-1}0^{m-1}10)$, and this is less than $\rho(1^{2n_0-1}0^{m+1}1)$ because $f_0 f_1 < f_1 f_0$. □

Now suppose $n_0 = 1$. We saw in the proof of Claim 2 above that $\rho(11) < \rho(00)$ implies $n_0 = 1$, but a priori both $\rho(11) < \rho(00)$ and $\rho(00) < \rho(11)$ are possible when $n_0 = 1$. Note that $\rho(10)$ is always minimal, however. The possible 3-minima are $\rho(1^3)$ (if $\rho(11)$ is maximal), $\rho(0^21)$ (if $\rho(00)$ is maximal), and $\rho(10^2)$ (in either case). But $\rho(1^3) > \rho(1^20)$ by $n_0 = 1$ and $\rho(0^21) > \rho(10^2)$ because $f_1 f_0^2 > f_0^2 f_1$, as observed in the base case of Claim 2. Hence $\rho(10^2)$ is always the 3-minimum when $n_0 = 1$. We now split into two final subcases to finish the argument when $x_0 > i_x$ and $n_0 = 1$. First, assume $\rho(00) < \rho(11)$, so $\rho(11)$ is maximal. For any $m \geq 2$, if $\rho(10^{m-1}1)$ is maximal and $\rho(10^m)$ is minimal, the next maximum is $\rho(10^m1)$ since the other possibility is $\rho(10^{m-1}10)$, which is of the form $f_0 f_1 y$ for some y, and $f_0 f_1 y < f_1 f_0 y$. And in this case the next minimum is $\rho(10^{m+1})$, because the other option is $\rho(10^{m-1}1^2)$, which is of the form $f_1^2 f_0 y$ hence greater than $f_0 f_1^2 y$. So by induction $\rho(10^m 1)$ is witnessed for all m if $\rho(11)$ is maximal.

The remaining subcase of $x_0 > i_x$ is $n_0 = 1$ and $\rho(11) < \rho(00)$. In general, it may be that $\rho(0^{\ell-1})$ is maximal for finitely many $\ell \geq 3$, but this cannot be the case for all ℓ (as we assume $b < 1$) because $\rho(0^\ell)$ decreases to r_0 as ℓ increases, while some probabilities of every length will be greater than r_1. Suppose $\rho(0^{\ell-1})$ is maximal and (by induction) $\rho(10^{\ell-2})$ is minimal, for $\ell \geq 3$. Then either $\rho(0^\ell)$ or $\rho(10^{\ell-2}1)$ is ℓ-maximal, and $\rho(10^{\ell-1})$ is always ℓ-minimal (by the same argument as in the last paragraph). If $\rho(10^{\ell-2}1)$ is maximal, then the argument in Claim 3 takes over from length $\ell+1$ onwards. If $\rho(0^\ell)$ is maximal, then $\rho(10^\ell)$ is $(\ell+1)$-minimal because the other option is $\rho(0^\ell 1) > \rho(0^{\ell-1}10)$. The argument then repeats for $\ell+1$, and so on, meaning we witness 0^ℓ for finitely many ℓ and then $10^m 1$ for all large enough m.

This completes the argument when $x_0 > i_x$. If instead $x_0 < i_x$, then something similar happens, but with even-odd parities switched. We state without detailed proofs the three claims (corresponding to those above) that will finish the argument here, as their proofs follow in the same way *mutatis mutandis.* First, there is a least n_0 such that $\rho(1^{2n_0+2}) > \rho(1^{2n_0}0^2)$. The case $n_0 = 0$ is separate and exactly analogous to the case $n_0 = 1$ when $x_0 > i_x$: if $n_0 = 0$ then $\rho(11) > \rho(00)$, so $\rho(00)$ is minimal, $\rho(01)$ is maximal, and in general we have 0^m minimal and $0^{m-1}1$ maximal for all $m \geq 2$. So assume $n_0 > 0$ from now on.

The second claim is that when $n_0 > 0$ and $x_0 < i_x$, for all $1 \leq \ell \leq n_0$, $\rho(1^{2\ell-2}01)$ is 2ℓ-maximal; $\rho(1^{2\ell})$ is 2ℓ-minimal; $\rho(1^{2\ell+1})$ is $(2\ell+1)$-maximal; and $\rho(1^{2\ell}0)$ is $(2\ell+1)$-minimal. Both the base case and the inductive step work very similarly as before (here, $\rho(11)$ being minimal relies on $n_0 > 0$). The only possible $(2\ell+2)$-maxima are $\rho(1^{2\ell}01)$ and $\rho(1^{2\ell+1}0)$; the only possible $(2\ell+2)$-minima are $\rho(1^{2\ell+2})$ and $\rho(1^{2\ell}0^2)$; the only possible $(2\ell+3)$-maxima are $\rho(1^{2\ell+3})$ and $\rho(1^{2\ell}010)$; and the only possible $(2\ell+3)$-minima are $\rho(1^{2\ell+2}0)$ and $\rho(1^{2\ell-2}01^2)$. All the alternatives listed can be dispensed with using $f_0 f_1 < f_1 f_0$, $f_0 f_1^2 < f_1^2 f_0$, and $\ell < n_0$. (The latter is only needed to show the claim holds for $\ell+1$ given it holds for ℓ, and so it does hold for $\ell = n_0$ as stated.)

The third and last claim needed is that when $n_0 > 0$ and $x_0 < i_x$, for all $m \geq 1$, we have $\rho(1^{2n_0}0^m)$ minimal and $\rho(1^{2n_0}0^{m-1}1)$ maximal. The case $m = 1$ follows by taking $\ell = n_0$ in the previous claim. The inductive step is again very similar to Claim 3 for $x_0 > i_x$, since the other possible minimum is $\rho(1^{2n_0}0^{m-1}1^2)$, which is of the form $f_1^2 f_0 y$ and hence not minimal since it is greater than $f_0 f_1^2 y$. The other possible maximum is $\rho(1^{2n_0}0^{m-1}10)$, of the form $f_0 f_1 y < f_1 f_0 y$, hence not maximal.

To finish the argument for Case 4(a), we address the special case when $b = 0$, i.e., f_0 is constant. (The subcase where instead f_1 is constant was dealt with in Case 2.) Here $r_0 = a$, and $\rho(w^\frown 01) = f_1 a$ for all strings w. If $x_0 > i_x$, then proceeding as above, we see that after applying f_1 some number of times, if n is least such that $\rho(1^{2n+1}) \geq a$, then it is not possible for the probability of any string with length greater than $2n$ to exceed $f_1 a$. This means that maximal probabilities cease to be unique at length $2n+1$, and only finitely many strings can be witnessed. The same holds when $x_0 < i_x$, but now the maxima cease to be unique after $\rho(1^{2n}) \geq a$.

Strings witnessed in the above case: $1^{2n-1}0^m1$, $1^{2n}0^m1$, 0^m, 0^m1.

(b) Both f_0 and f_1 increase i_x. This is equivalent to $f_0 f_1 x > f_1 f_0 x$ for all x, so that appending 10 always gives a higher probability than appending 01. It is also equivalent to $f_0 f_1^2 x > f_1^2 f_0 x$ for all x (see Lemma 4.4). As before, we put off the special case $b = 0$ for later, and assume for the moment that $b > 0$.

First, say that $x_0 < i_x$. Here we need to split into slightly different subcases than we did in (a). Since $\rho(0) < \rho(1)$, the 2-maximum is always $\rho(10)$ because the only other option is $\rho(01) < \rho(10)$. The possible 2-minima are $\rho(00)$ and $\rho(11)$, and both $\rho(00) < \rho(11)$ and $\rho(11) < \rho(00)$ are possible. Suppose first that $\rho(00) < \rho(11)$. It may be that $\rho(0^\ell)$ is minimal for finitely many ℓ, but eventually this is no longer the case since $\rho(0^\ell)$ increases to r_0 while some other probabilities always stay below r_1. Suppose that for some $\ell \geq 2$, $\rho(0^\ell)$ is minimal and $\rho(10^{\ell-1})$ is maximal. The possible $(\ell+1)$-maxima are $\rho(10^\ell)$ and $\rho(0^\ell 1)$, but the latter is of the form $f_1 f_0 y$ for some y, which is less than $f_0 f_1 y$ and so not maximal. The possible $(\ell+1)$-minima are $\rho(0^{\ell+1})$ and $\rho(10^{\ell-1}1)$. Either may be the case in general, and if $\rho(0^{\ell+1})$ is minimal then the argument repeats for length $\ell+1$: now $\rho(10^\ell)$ is maximal. For large enough ℓ, that is no longer the case, and for such an ℓ we have $\rho(10^{\ell-1})$ maximal and $\rho(10^{\ell-2}1)$ minimal. Once that happens, the $(\ell+1)$-maximum is $\rho(10^\ell)$ since $\rho(10^{\ell-2}1^2)$ is of the form $f_1^2 f_0 y$ for some y, which is less than $f_0 f_1^2 y$. The $(\ell+1)$-minimum is $\rho(10^{\ell-1}1)$ since the other option is $\rho(10^{\ell-2}10)$, which is of the form $f_0 f_1 y$, and this is greater than $f_1 f_0 y$. It follows by induction that we witness 10^m for all $m \geq 0$ in this case.

For the rest of the argument for $x_0 < i_x$, we assume instead that $\rho(11) < \rho(00)$. The argument follows from a series of three claims, much like in part (a). First, there is a least n_0 such that $\rho(1^{2n_0-1}0^2) > \rho(1^{2n_0+1})$. Then $n_0 \geq 1$. The case $n_0 = 1$ requires special treatment, which we outline before proceeding further. We have $\rho(11)$ minimal and $\rho(10)$ maximal. Since $\rho(1^3) < \rho(10^2)$ when $n_0 = 1$, the 3-maximum is $\rho(10^2)$, and the 3-minimum is $\rho(101)$ because the other option $\rho(1^2 0)$ is greater than $\rho(01^2)$. Inductively, if for $m \geq 2$ we have $\rho(10^m)$ maximal and $\rho(10^{m-1}1)$ minimal, then the $(m+2)$-maximum is $\rho(10^{m+1})$ since the other option, $\rho(10^{m-1}1^2)$, is of the form $f_1^2 f_0 y < f_0 f_1^2 y$, so not maximal. And the $(m+2)$-minimum is $\rho(10^m 1)$ since the other option is $\rho(10^{m-1}10)$, which is of the form $f_0 f_1 y > f_1 f_0 y$, so not minimal. It follows that we witness 10^m for all $m \geq 0$ in this subcase.

Now assume $n_0 > 1$ as well as $\rho(11) < \rho(00)$. The second claim to complete the proof is that for any $1 \leq \ell \leq n_0$, we have that $\rho(1^{2\ell-1}0)$ is 2ℓ-maximal; $\rho(1^{2\ell})$ is 2ℓ-minimal; if $\ell < n_0$, then $\rho(1^{2\ell+1})$ is $(2\ell+1)$-maximal; and $\rho(1^{2\ell-1}01)$ is $(2\ell+1)$-minimal. The induction argument goes exactly as in Claim 2 from case (a) where $x_0 > i_x$, except switching the roles of "maximal" and "minimal" everywhere as well as switching the roles of (firstly) $f_0 f_1$ and $f_1 f_0$, and (secondly) $f_0 f_1^2$ and $f_1^2 f_0$. This is because we now have $f_1 f_0 < f_0 f_1$ and $f_1^2 f_0 < f_0 f_1^2$ by Lemma 4.4. The third claim, which completes the picture, is that $\rho(1^{2n_0-1}0^m)$ is maximal and $\rho(1^{2n_0-1}0^{m-1}1)$ is minimal for all $m \geq 0$. The cases $m = 0$ and $m = 1$ follow from taking $\ell = n_0 - 1$ and $\ell = n_0$ in the second claim. For $m = 2$, the $(2n_0+1)$-minimum is $\rho(1^{2n_0-1}01)$ by the second claim again, and the $(2n_0+1)$-maximum is $\rho(1^{2n_0-1}0^2)$ because the other option is $\rho(1^{2n_0+1})$, and this is the lesser value by definition of n_0. The induction can be carried out from here using $f_1^2 f_0 < f_0 f_1^2$ and $f_1 f_0 < f_0 f_1$, finishing the proof for $x_0 < i_x$.

Now suppose $x_0 > i_x$. The proof of this case is split into three claims, as usual. First, there is a least n_0 such that $\rho(1^{2n_0+2}) \leq \rho(1^{2n_0}0^2)$. As before, we first need to consider the case $n_0 = 0$ separately, but fortunately this is equivalent to $\rho(11) < \rho(00)$ so there is no need for a third subcase as with the $x_0 < i_x$ argument. If $n_0 = 0$, then $\rho(00)$ is maximal since $\rho(1) < \rho(0)$, and $\rho(01)$ is minimal by $f_1f_0 < f_0f_1$. In general, suppose for any $m \geq 2$ that $\rho(0^m)$ is m-maximal and $\rho(0^{m-1}1)$ is m-minimal. Then $\rho(0^{m+1})$ is $(m+1)$-maximal since the other option is $\rho(0^{m-1}1^2)$, which is of the form $f_1^2f_0y < f_0f_1^2y$ and so not maximal. And $\rho(0^m1)$ is $(m+1)$-minimal since the other option $\rho(0^{m-1}10)$ is of the form $f_0f_1y > f_1f_0y$, hence not minimal. It follows by induction that 0^m is witnessed in this subcase for all $m \geq 1$.

Assume from now on that instead $n_0 > 0$. The second claim we need to finish the proof is that for $1 \leq \ell \leq n_0$, $\rho(1^{2\ell})$ is 2ℓ-maximal; $\rho(1^{2\ell-2}01)$ is 2ℓ-minimal; $\rho(1^{2\ell}0)$ is $(2\ell+1)$-maximal; and $\rho(1^{2\ell+1})$ is $(2\ell+1)$-minimal. The base case here uses $n_0 > 0$ to show $\rho(11)$ is maximal. The third claim is that $\rho(1^{2n_0}0^m)$ is maximal and $\rho(1^{2n_0}0^{m-1}1)$ is minimal for all $m \geq 2$. Here, for $m = 2$, we have $\rho(1^{2n_0}0^2)$ maximal since by definition of n_0, $\rho(1^{2n_0+2})$ cannot be. And $f_0f_1 > f_1f_0$ implies that $\rho(1^{2n_0}01)$ is minimal rather than $\rho(1^{2n_0+1}0)$. The inductive steps of both claims can be shown in a straightforward way using $f_1f_0 < f_0f_1$, $f_1^2f_0 < f_0f_1^2$, and in the first statement of the second claim, $\ell < n_0$. (The latter is used only to show the second claim holds for $\ell+1$ given it holds for ℓ, so it does hold for $\ell = n_0$ as stated.)

Finally, suppose $b = 0$. As in case (a), only finitely many strings can be witnessed. We have again that $r_0 = a$ and $\rho(w^\frown 01) = f_1a$ for all strings w. If $x_0 < i_x$, and n is large enough that $\rho(1^{2n+1}) \leq a$, then no longer string can have probability greater than f_1a, and this value is never attained uniquely. If $x_0 > i_x$, and n is large enough that $\rho(1^{2n}) \leq a$, the same conclusion holds. Therefore at most finitely many constant strings can be witnessed, and nothing else. This completes the proof of Case 4(b) and of Theorem 4.2.

Strings witnessed in the above case: $1^{2n-1}0^m$, $1^{2n}0^m$.

4.3. **Proof of Theorem** 4.3. We show that for every string w listed in Theorem 4.1, there is an IFS (f_0, f_1, x_0) which falls into the subcase of the proof of Theorem 4.2 which would lead to w being witnessed. This results in a case breakdown into the following seven subfamilies of strings, listed here with the subcases of Theorem 4.2 which they employ:

- 0^n1^m for a given n and all m – Case 2(a), Proposition 4.5;
- $1^{2n}0^m1$ for a given n and all m – Case 4(a), Proposition 4.6;
- $1^{2n-1}0^m1$ for a given n and all m – Case 4(a), Proposition 4.7;
- $1^{2n}(01)^m$ for a given n and all m – Case 3(a), Proposition 4.8;
- $1^{2n+1}(01)^m$ for a given n and all m – Case 3(a), Proposition 4.9;
- $1^{2n+1}0(10)^m$ for a given n and all m – Case 3(b), Proposition 4.10;
- $0^{2n}1(01)^m$ for a given n and all m – Case 3(a), Proposition 4.11.

The proofs all follow the same basic strategy, which goes roughly as follows. Given n, derive an inequality equivalent to the condition from the relevant subcase of the proof of Theorem 4.2 which results in strings with prefixes of length n being witnessed. This translates to a requirement that x_0 be chosen inside a certain interval depending on n and the coefficients of the IFS. For any fixed n, a, b, c, d, finitely many of these intervals will overlap $[0,1]$, and the set of such intervals is closed downward in n: for any $\ell \leq n$, if n's interval overlaps $[0,1]$, so does ℓ's. Derive an inequality $n < f(a,b,c,d)$ for some function f which is equivalent to n's interval overlapping $[0,1]$. Treat three out of a,b,c,d as functions of the fourth and show that $f \to \infty$ as the fourth number tends to 1 or -1, depending on the subcase. This finishes the proof since it shows a,b,c,d can be chosen to satisfy $n < f$ for infinitely many n, which is enough.

Although all seven subcases follow this outline, the particularities are different enough to warrant separate treatments, albeit with some details omitted.

Proposition 4.5. *$A_P(0^n1^m) = 2$ for all $n, m \geq 0$.*

Proof. Let $n \geq 1$ be given (the case $n = 0$ is trivial). The IFS (f_0, f_1, x_0) witnesses 0^n1^m for all m if in Case 2(a) of the proof of Theorem 4.2 with $x_0 > i_x$, and if n is least such that $f_0^n x_0 < i_x$, i.e., $f_0^n x_0 < i_x < f_0^{n-1}x_0$. Take $f_0 = bx$ for $b < 1$ and $f_1 = c$ (so $a = d = 0$). Then $f_0^n x_0 = b^n x_0$, and our condition becomes

$$b^n x_0 < i_x < b^{n-1}x_0 \quad \text{or equivalently} \quad \frac{i_x}{b^{n-1}} < x_0 < \frac{i_x}{b^n}.$$

Since $b < 1$, we have $i_x/b^n > i_x$ for all $n \geq 1$. In order to choose x_0 to witness 0^n1^m for our given n, we need $i_x/b^{n-1} < 1$, or equivalently

$$\frac{\log i_x}{\log b} + 1 > n.$$

By increasing b arbitrarily close to 1, and setting $c = b/2$ from b, we can make $\log i_x / \log b$ larger than any given n, so that it is possible to choose $x_0 \in (i_x, 1)$ in order for exactly 0^n1^m to be witnessed for all $m \geq 0$. □

Proposition 4.6. *$A_P(1^{2n}0^m1) = 2$ for all $n, m \geq 0$.*

Proof. Let $n \geq 1$ be given ($n = 0$ is covered by the previous proposition). The IFS (f_0, f_1, x_0) witnesses $1^{2n}0^m1$ for all $m \geq 0$ if in Case 4(a) of the proof of Theorem 4.2—that is, $b > 0 > d$ and both maps decrease i_x—if $x_0 < i_x$, and if n is least such that

$$f_1^{2n+2}x_0 > f_0^2 f_1^{2n} x_0.$$

Thinking of the LHS here as $f_1^2 f_1^{2n} x_0$, this inequality is equivalent to

$$a + ab + b^2 f_1^{2n} x_0 < c + cd + d^2 f_1^{2n} x_0 \quad \iff \quad F := \frac{f_0^2 0 - f_1^2 0}{d^2 - b^2} < f_1^{2n} x. \tag{5}$$

If n is supposed to be the least number making $F < f_1^{2n}x_0$, then we would like $f_1^{2(n-1)}x_0 < F < f_1^{2n}x_0$. Note that for any x and n, we have

$$f_1^n x = c\sum_{i=0}^{n-1} d^i + d^n x = c \cdot \frac{1-d^n}{1-d} + d^n x = r_1(1-d^n) + d^n x,$$

and an analogous formula for $f_0^n x$. Then

$$f_1^{2(n-1)}x_0 < F \iff r_1(1 - d^{2(n-1)}) + d^{2(n-1)}x_0 < F \iff x_0 < r_1 - \frac{r_1 - F}{d^{2(n-1)}},$$

and on the other hand

$$F < f_1^{2n}x_0 \iff x_0 > r_1 - \frac{r_1 - F}{d^{2n}}$$

by a similar calculation. Now, in our situation it will always be the case that $F < r_1 = c/(1-d)$, because

$$\frac{f_0^2 0 - f_1^2 0}{d^2 - b^2} < \frac{c}{1-d} \iff a(1-d+b-bd) < c(1-b^2) \iff \frac{a}{1-b} < \frac{c}{1-d},$$

i.e., $r_0 < r_1$. As long as we choose a, b, c, d to make $r_0 < r_1$, then, we have $F < r_1$. We also need $F > 0$, but this will be guaranteed by $f_1^{2(n-1)} x_0 < F$ since the latter LHS is nonnegative for every $n \geq 1$ and x_0.

Overall, then, the IFS witnesses $1^{2n}0^m 1$ when we can pick x_0 such that

$$x_0 \in \left(r_1 - \frac{r_1 - F}{d^{2n}}, r_1 - \frac{r_1 - F}{d^{2(n-1)}} \right), \tag{6}$$

a nonempty interval since $d^{2n} < d^{2(n-1)}$ and $r_1 - F > 0$. Both endpoints of this interval are less than i_x if $r_0 < r_1$, since $r_0 < r_1$ iff $r_1 < i_x$, and so choosing such an x_0 automatically fulfills the requirement that $x_0 < i_x$.

We also need (for a given n) to be able to pick $x_0 > 0$, so at least the right endpoint in (6) should be positive. For any n.

$$r_1 - \frac{r_1 - F}{d^{2(n-1)}} > 0 \iff d^{2(n-1)} > 1 - \frac{F}{r_1} \iff (n-1)\log d^2 > \log\left(1 - \frac{F}{r_1}\right)$$
$$\iff n < 1 + \frac{\log(1 - F/r_1)}{\log d^2}.$$

For arbitrarily large n to be possible, the last RHS must be able to grow arbitrarily large depending on a, b, c, d. To accomplish this we can treat d as a variable and make c a function of d (so that F and r_1 are as well), then require that

$$\lim_{d \to -1^+} \frac{\log(1 - F/r_1)}{\log d^2} = \infty. \tag{7}$$

We need c to be a function of d because c must be greater than $|d|$ for all $d > -1$ if we are to have $c + d > 0$, so c will necessarily approach 1 in the limit. Of course we also need to make sure the logarithm in the numerator is defined for all $d > -1$. If so, then together with the fact that the right endpoint of (6) is always less than i_x, we will have that for *every* $n \geq 1$ there is a choice of a, b, c, d, x_0 making (f_0, f_1, x_0) witness $1^{2n}0^m 1$ for all m.

So, to sum up thus far, we want to choose numbers a, b and a continuous function $c(d)$ to satisfy the requirements that $|d| < c(d) < 1$, $a < 1 - b$, $r_0 < r_1$, $i_x < 1$, $i_y > 0$, and the limit condition (7) holds. Because we are taking the limit as $d \to -1^+$, we may as well only bother asking for the other requirements to hold in the limit, too. This simplifies things considerably: since $c \to 1$ as $d \to -1$, we have $r_1 \to 1/2$. Then for $r_0 < r_1$ to hold in the limit, it is enough to make $r_0 = a/(1-b) < 1/2$, or in other words $2a < 1 - b$. This condition also guarantees $a < 1 - b$ and hence $a + b \in [0, 1]$. Furthermore, since $c(d)$ will eventually be greater than any fixed $a < 1$, $a < c$ is satisfied in the limit. That $c + d \in [0, 1]$ is implied by the requirement that $|d| < c(d) < 1$.

Only two conditions remain to be checked. Firstly, (7) holds if $1 - F/r_1$ stays strictly between 0 and 1 as $d \to -1^+$: on the one hand, $0 < 1 - F/r_1$ iff $F < r_1$, which as we saw is equivalent to $r_0 < r_1$. On the other hand, $1 - F/r_1 < 1$ iff both F and r_1 are positive, and both of those happen in the limit as noted above. Finally, we need to check that the lines intersect in $[0, 1]^2$. But since $f_1(x) \to 1 - x$

as $d \to -1$, if we make sure to take $a, b > 0$, then f_1 will eventually intersect any line that stays inside $[0,1]^2$. Hence $i_y > 0$ and $i_x < 1$ hold in the limit, and we are done. □

The proofs of all but one of the remaining cases are very similar to the above, and we will give a somewhat more streamlined presentation from here on out. The most complicated subcase we save for last (Proposition 4.11).

Proposition 4.7. *$A_P(1^{2n-1}0^m 1) = 2$ for all $n \geq 1$, $m \geq 0$.*

Proof. If $n \geq 1$ is given, then (f_0, f_1, x_0) witnesses $1^{2n-1}0^m 1$ for all $m \geq 0$ if in Case 4(a) of the proof of Theorem 4.2 (mixed slopes) with $x_0 > i_x$ and n least such that

$$f_1^{2n+1}x_0 > f_0^2 f_1^{2n-1}x_0. \tag{8}$$

As before, we will pick $a, b > 0$ constants and c a continuous function of d so that as $d \to -1^+$, it is always possible to choose x_0 making the above happen for any given n. Now, (8) is equivalent to

$$f_1^{2n-3}x_0 < E < f_1^{2n-1}x_0 \quad \text{where} \quad E = \frac{a(1+b) - c(1+d)}{d^2 - b^2}, \tag{9}$$

and the inequality in (9) is equivalent to

$$x_0 \in \left(r_1 + \frac{r_1 - E}{|d|^{2n-3}}, r_1 + \frac{r_1 - E}{|d|^{2n-1}} \right). \tag{10}$$

For arbitrarily large n to be possible, we want to pick a, b, c, d so that this interval intersects $(i_x, 1)$, so a suitable x_0 can be chosen. The left endpoint in (10) can be made less than 1 for arbitrarily large n if, in particular,

$$\lim_{d \to -1^+} \frac{\log \dfrac{r_1 - E}{1 - r_1}}{\log d^2} + \frac{3}{2} = \infty. \tag{11}$$

And the right endpoint in (10) is greater than i_x, for a given n, iff

$$\frac{\log \dfrac{r_1 - E}{i_x - r_1}}{\log d^2} + \frac{1}{2} > n. \tag{12}$$

Note that in the limit, E approaches r_0 (as long as $b < 1$). Hence as long as $r_0 < r_1$ in the limit, then eventually $r_1 > E$. If we arrange things so i_x stays below 1, then,

$$\frac{r_1 - E}{i_x - r_1} > \frac{r_1 - E}{1 - r_1}.$$

Also notice that we can take $\frac{r_1 - E}{i_x - r_1} < 1$ in the limit since this is equivalent to $2r_1 < i_x + E$, which in the limit is guaranteed if $2a + b < 1$, as may be checked with a little algebra. Assume that a, b are positive with $2a + b < 1$. Then since log is increasing, if (11) holds, the LHS of (12) will also approach ∞. This implies that whenever n is such that the left endpoint of (10) is less than 1, for all $n' \leq n$ it is possible to choose $x_0 \in (i_x, 1)$ in order to witness $1^{2n'-1}0^m 1$.

So, let $c(d)$ be a continuous function with $|d| < c(d) < 1$ for all $d > -1$, and let $a, b > 0$ be such that $2a+b < 1$. This immediately implies $a+b, c+d \in [0,1]$ for all d. Since $r_1 \to 1/2$ as $d \to -1$, we have $r_0 < r_1$ in the limit since $r_0 = a/(1-b) < 1/2$. We also need $E > 0$, which is guaranteed as $d \to -1$ since E approaches $r_0 > 0$.

Since $c \to 1$ and $d \to -1$, eventually $c > a$ as required. Because $f_1(x) \to 1 - x$ as $d \to -1$, $c + dx$ will eventually intersect $a + bx$ in $[0,1]^2$, so that $0 < i_y < i_x < 1$. It only remains to check (11). But we already observed that

$$\frac{r_1 - E}{1 - r_1} < \frac{r_1 - E}{i_x - r_1} < 1$$

as $d \to -1$, and $\frac{r_1 - E}{1 - r_1} > 0$ iff $r_1 > E$, which also holds in the limit. Therefore the logarithm in the numerator approaches a finite negative number, while $\log d^2$ approaches 0 from below. □

Proposition 4.8. *$A_P(1^{2n}(01)^m) = 2$ for all $n, m \geq 0$.*

Proof. For a given n, we witness $1^{2n}(01)^m$ if in Case 3(a) of the proof of Theorem 4.2 (both slopes negative), with $x_0 > i_x$ and n least such that

$$f_1^{2n} x_0 < i_x. \tag{13}$$

We will pick numbers $a > 0$, $b < 0$, and a continuous function $c(d)$ so that as $d \to -1^+$, we have $a + b, c + d \in [0,1]$, $a < c$, $b > d$, $r_0 < r_1$, and the lines $a + bx$ and $c + dx$ intersecting in $[0,1]^2$. If $a, b \notin \{0, \pm 1\}$, then the last condition is automatically met as $d \to -1$ since $f_1 \to 1 - x$ and this intersects any line in $[0,1]^2$. The conditions $a < c$ and $b > d$ are also automatically met as $d \to -1$. At the same time, we must (given n) be able to pick

$$x_0 \in \left(r_1 + \frac{i_x - r_1}{d^{2(n-1)}}, r_1 + \frac{i_x - r_1}{d^{2n}} \right) \tag{14}$$

so that $f_1^{2n} x_0 < i_x < f_1^{2(n-1)} x_0$. We need this interval to intersect $(i_x, 1)$ for arbitrarily large n, for suitable choices of a, b, c, d. That the right endpoint is always greater than i_x, for any n, follows from $d^{2n} < 1$, since then $\frac{i_x - r_1}{d^{2n}} > i_x - r_1$. For the left endpoint to be less than 1 for arbitrarily large n we need

$$\lim_{d \to -1^+} \frac{\log \dfrac{i_x - r_1}{1 - r_1}}{\log d^2} = \infty. \tag{15}$$

Pick $a > 0 > b$ with

$$-b < a < \frac{1 - b}{2}. \tag{16}$$

Also let $c(d)$ be a continuous function with $|d| < c(d) < 1$ for all $d > -1$. This immediately gives $c + d \in [0,1]$, and (16) implies $a + b \in [0,1]$ too. Next, since $r_1 \to 1/2$ as $d \to -1$ and (16) makes $r_0 = a/(1 - b) < 1/2$, we have $r_0 < r_1$ in the limit. Finally, to satisfy (15), we want $\frac{i_x - r_1}{1 - r_1}$ to be strictly between 0 and 1 in the limit. This quantity is automatically positive since $i_x > r_1$ and $1 > r_1$ (both in the limit, again). And because (16) implies $i_x \to \frac{1-a}{b+1} < 1$ as $d \to -1$, the fraction is also less than 1 in the limit. This completes the proof. □

Proposition 4.9. *$A_P(1^{2n+1}(01)^m) = 2$ for all $n, m \geq 0$.*

Proof. Given n, take the IFS to be in Case 3(a) of the proof of Theorem 4.2 (both slopes negative) with $x_0 < r_0$ and n such that

$$f_1^{2n+1} x_0 < i_x < f_1^{2n-1} x_0. \tag{17}$$

This is equivalent to

$$x_0 \in \left(r_1 - \frac{i_x - r_1}{|d|^{2n+1}}, r_1 - \frac{i_x - r_1}{|d|^{2n-1}}\right). \tag{18}$$

Since

$$r_1 - \frac{i_x - r_1}{|d|^{2n+1}} < r_0 \iff |d|^{2n+1} < \frac{i_x - r_1}{r_1 - r_0}$$

and the last fraction is greater than 1 by Lemma 4.4(e) while the LHS is less than 1, we have that the left endpoint of (18) is always less than r_0 for all $n \geq 0$. In order to make the right endpoint of (18) greater than 0 for arbitrarily large n (for suitable choice of a, b, c, d), so that an $x_0 \in (0, r_0)$ may be chosen to make the IFS witness exactly $1^{2n+1}(01)^m$, we can arrange for

$$\lim_{d \to -1^+} \frac{\log(i_x/r_1 - 1)}{\log d^2} = \infty. \tag{19}$$

As usual, pick constants $a, b \notin \{0, \pm 1\}$, $a > 0 > b$, and a continuous function $c(d)$ such that $|d| < c(d) < 1$ for all $d > -1$ (so $c + d \in [0,1]$). To satisfy (19), we want $0 < i_x/r_1 - 1 < 1$ in the limit, or equivalently $r_1 < i_x < 2r_1$. Since i_x converges to $(1-a)/(b+1)$ and $r_1 \to 1/2$, this can achieved (along with $a + b \in [0,1]$) by making $-b < a < \frac{1-b}{2}$. This implies that $a < c$ and $b > d$ are met in the limit, and again since $f_1 \to 1 - x$ we will eventually have $(i_x, i_y) \in [0,1]^2$. And $r_0 < r_1$ follows from $r_1 < i_x$. □

Proposition 4.10. $A_P(1^{2n+1}0(10)^m) = 2$ *for all* $n, m \geq 0$.

Proof. For this, given n, we take the IFS to be in Case 3(b) of the proof of Theorem 4.2, so that both maps have negative slope and *increase* i_x. We want $x_0 > r_0$ and n to be such that

$$f_1^{2n-1} x_0 < i_x < f_1^{2n+1} x_0.$$

This is equivalent to

$$x_0 \in \left(r_1 + \frac{r_1 - i_x}{|d|^{2n-1}}, r_1 + \frac{r_1 - i_x}{|d|^{2n+1}}\right). \tag{20}$$

Remember that in the present case we have $i_x < r_1 < r_0$. We will pick $a > 0 > b$ with

$$\frac{1-b}{2} < a < 1, \tag{21}$$

and pick $c(d)$ a continuous function with $|d| < c(d) < 1$ for all $d > -1$. Then if we take $d \to -1$, we have $r_1 \to 1/2$ and $r_0 > 1/2$ by choice of a and b, so that $r_1 < r_0$ in the limit. Also $a + b, c + d \in [0,1]$, $a < c$, and $b > d$ hold in the limit; and as before, $a + bx$ eventually intersects $c + dx$ in $[0,1]^2$ since $c + dx \to 1 - x$. Now we just need to make sure we can always pick an $x_0 \in (r_0, 1)$ for arbitrarily large n as $d \to -1$. We have

$$r_1 + \frac{r_1 - i_x}{|d|^{2n+1}} > r_0 \iff \frac{r_1 - i_x}{r_0 - r_1} > |d|^{2n+1}.$$

Since the RHS here is less than 1 and the LHS is greater than 1 (by Lemma 4.4(e) again), this always happens for any n. To make the left endpoint of (20) less than

1 for any given n, so that suitable a, b, c, d, x_0 may be chosen to witness the desired string, it suffices to ensure that

$$\lim_{d \to -1^+} \frac{\log \dfrac{r_1 - i_x}{1 - r_1}}{\log d^2} = \infty. \tag{22}$$

Thus we want $0 < \frac{r_1 - i_x}{1 - r_1} < 1$ in the limit, or equivalently $2r_1 - 1 < i_x < r_1$. Since $r_1 \to 1/2$, in the limit the latter becomes

$$0 < \frac{1-a}{b+1} < \frac{1}{2},$$

which is equivalent to (21). □

Now we arrive at the final and most complex subcase of Theorem 4.3 to prove. The extra difficulty arises because, basically, we will need to take both b and d to -1 while both a and c go to 1. This makes it harder to make certain properties hold "in the limit" as in the previous subcases, and also results in a limit condition in which the limit converges to $\log \frac{0}{0}$. Slightly more delicate handling is needed to get around these issues.

Proposition 4.11. *$A_P(0^{2n}1(01)^m) = 2$ for all $n, m \geq 0$.*

Proof. Let n be given. The IFS (f_0, f_1, x_0) witnesses $0^{2n}1(01)^m$ for all m if in Case 3(a) of the proof of Theorem 4.2 (where both maps have negative slope and both decrease i_x), when $x_0 > i_x$ and when n is least such that $f_0^{2n}x_0 < i_x$, i.e.,

$$f_0^{2n}x_0 < i_x < f_0^{2(n-1)}x_0,$$

or equivalently (after rearranging)

$$x_0 \in \left(r_0 + \frac{i_x - r_0}{b^{2n-2}}, r_0 + \frac{i_x - r_0}{b^{2n}}\right). \tag{23}$$

If we can pick a, b, c, d to make $r_0 < i_x$, then this interval is nonempty with positive endpoints. For this n and a, b, c, d, it is possible to choose x_0 to witness the desired family of strings iff the left endpoint is less than 1 and the right endpoint is greater than i_x. First,

$$r_0 + \frac{i_x - r_0}{b^{2n}} > i_x \iff 1 > b^{2n},$$

which is true for all $n \geq 1$, so if an x_0 can be chosen above i_x for a given n then a suitable x_0 can also be chosen for any $n' \leq n$. And we can choose $x_0 < 1$ iff

$$r_0 + \frac{i_x - r_0}{b^{2n-2}} < 1 \iff \frac{i_x - r_0}{1 - r_0} < b^{2n-2} \iff \frac{\log \dfrac{i_x - r_0}{1 - r_0}}{\log b^2} + 1 > n.$$

This is possible to achieve for any given n if we can make

$$\lim_{b \to -1^+} \frac{\log \dfrac{i_x - r_0}{1 - r_0}}{\log b^2} = \infty. \tag{24}$$

Altogether this means that if $r_0 + (i_x - r_0)/b^{2n} < 1$ for some n and a fixed choice of a, b, c, d, then it is possible for every $1 \leq n' \leq n$ to pick a suitable value of $x_0 > i_x$ making (f_0, f_1, x_0) witness the strings $0^{2n'}1(01)^m$ for every $m \geq 0$. Hence the proof will be complete if we can choose a, c, and d as functions of b such that such that (24) holds and such that the IFS remains in Case 3(a) of the proof of Theorem 4.2 for all $b > -1$. Actually, for technical reasons it will be simpler for now to choose r_0 as a function of b and then let $a(b) = (1-b)r_0(b)$. This is not a problem because b is never 1, so $r_0(b) = a(b)/(1-b)$ is always well-defined. We will ultimately see that the requirements we impose on $r_0(b)$ do not contradict the behavior of $a(b)$.

We proceed by deriving necessary conditions on r_0, c, d to satisfy each requirement, and showing along the way that each new condition is compatible with all the preceding ones. This will imply that functions r_0, c, d satisfying all of them do indeed exist. Our first requirements, which we will take as "atomic" in that they will not reduce to other requirements, are that

$$(1-b)r_0(b) < 1 \quad \text{and} \quad |b| < |d(b)| < c(b) < 1 \tag{25}$$

for all $b > -1$ (with b, d negative). The second of these immediately implies $c + d \in [0,1]$. To guarantee $a + b \in [0,1]$, first note that $a + b = r_0(1-b) + b < 1$ iff $r_0 < 1$, and this follows from the first atomic requirement. Then $a + b > 0$ iff

$$r_0 > -b/(1-b), \tag{26}$$

a new requirement. Actually, (26) will turn out to be a consequence of $i_x, i_y \in [0,1]$, or in other words of f_0 and f_1 intersecting in $[0,1]^2$. We need the latter to happen anyway, so let us now find a sufficient condition for it. Rewriting i_x and i_y in terms of r_0 produces

$$i_x = \frac{c - r_0(1-b)}{b-d} \quad \text{and} \quad i_y = \frac{bc - r_0(1-b)d}{b-d}.$$

If $i_y < i_x$, or equivalently $r_0 < r_1$, then it suffices to make $i_y > 0$ and $i_x < 1$. We will address the issue of ensuring $r_0 < r_1$ in a moment. One can check that

$$i_y > 0 \iff r_0 > \frac{bc}{d(1-b)} \quad \text{and} \quad i_x < 1 \iff r_0 > \frac{c+d-b}{1-b}. \tag{27}$$

Since $c + d > 0$, we have $\frac{c+d-b}{1-b} > \frac{-b}{1-b}$, so that satisfying (27) would automatically result in (26) being satisfied too. Thus (26) is redundant. Next, some more algebra shows that

$$\frac{bc}{d(1-b)} < \frac{c+d-b}{1-b} \iff b > d,$$

an atomic requirement. Hence the first condition in (27) is implied by the second as long as (25) holds, so is also redundant. Then we will have $a + b > 0$, $i_y > 0$, and $i_x < 1$ if we can choose r_0 so that

$$\frac{c+d-b}{1-b} < r_0 < \frac{c}{1-d} = r_1. \tag{28}$$

The latter guarantees that $i_y < i_x$ so that we stay in Case 3(a) of the proof of Theorem 4.2, and also subsumes the second condition in (27), so if (28) holds then (27) is fully redundant. Now, the interval in (28) is nonempty because

$$\frac{c+d-b}{1-b} < \frac{c}{1-d} \iff (c+d-b)(1-d) < c(1-b) \iff (c-1+d)(b-d) < 0,$$

which follows from the second requirement in (25): $b - d > 0$ since $b > d$, and $c - 1 + d < 0$ since $|d| < c < 1$. So (25) makes it possible to choose r_0 to satisfy (28), and together (25) and (28) are enough to ensure we stay in Case 3(a).

It remains to show that the limit requirement (24) is consistent with (25) and (28). We will take r_0, c, and d to be continuously differentiable functions of b. $\log b^2$ approaches 0 from below as $b \to -1^+$, so in order for the limit to reach $+\infty$, one needs the logarithm in the numerator to stay negative. For this, one wants to maintain

$$0 < \frac{i_x - r_0}{1 - r_0} < 1$$

in the limit, and for this quantity to stay strictly below 1 at $b = -1$. Now, $d(b) \to -1^+$ as $b \to -1^+$ since d is always less than b, and $c(b) \to 1$. Then after yet more algebra, we have that

$$\frac{i_x - r_0}{1 - r_0} = \frac{c - r_0(1-d)}{(b-d)(1-r_0)} \to \frac{0}{0} \quad \text{as } b \to -1.$$

An application of L'Hôpital's Rule shows that the limit is equal to

$$\lim_{b \to -1^+} \frac{c' - r_0'(1-d) + r_0 d'}{(1-r_0)(1-d') - r_0'(b-d)} = \frac{2c'(-1) - 4r_0'(-1) + d'(-1)}{1 - d'(-1)}. \tag{29}$$

(The calculation follows since r_0', c', and d' are bounded everywhere by assumption, and $r_0 \to 1/2$.) Since d decreases to -1 as b decreases to -1, $d'(-1) \geq 0$, and we will need $d'(-1) \neq 1$ for (29) to be well-defined. If we take $0 < d'(-1) < 1$, then the denominator of the limit in (29) is positive. Hence the limit in (24) will tend to $\frac{-\infty}{0^-} = +\infty$, as needed, if

$$0 < \frac{2c'(-1) - 4r_0'(-1) + d'(-1)}{1 - d'(-1)} < 1. \tag{30}$$

If $L(b) = \frac{c+d-b}{1-b}$ is the lower bound in (28), then one can calculate

$$L'(-1) = \frac{2c'(-1) + 2d'(-1) - 1}{4}, \quad r_0'(-1) = \frac{2a'(-1) + 1}{4},$$
$$\text{and} \quad r_1'(-1) = \frac{2c'(-1) + d'(-1)}{4}.$$

Using these expressions we see that (30) is equivalent to

$$L'(-1) < r_0'(-1) < r_1'(-1). \tag{31}$$

Our final objective is to show (31) is consistent with the other requirements (25) and (28), which will complete the proof since that means (24), (25), and (28) can all be satisfied simultaneously. Actually, under the above assumption that $0 < d'(-1) < 1$, and up to possibly perturbing r_0, c, and d, (31) is equivalent to (28) holding in the limit. This follows because for any continuously differentiable functions $f(x), g(x)$ having the same limit as $x \to C^+$, where C is some constant, then for any $\varepsilon > 0$, $f(x) > g(x)$ on $(C, C+\varepsilon)$ iff $f'(x) > g'(x)$ on $(C, C+\varepsilon)$. Then since L, r_0, and r_1 all tend to $1/2$ as $b \to -1$, we have that (28) holding in a right neighborhood of $b = -1$ is equivalent to $L' < r_0' < r_1'$ holding in the same neighborhood. By smoothly perturbing r_0, c, and d if necessary, as long as $0 < d'(-1) < 1$ is maintained, we can ensure strict inequality between the derivatives holds at $b = -1$, i.e., that (31) holds. (A bit more formally, one could say that these strict inequalities are all open conditions in the C^1 topology.) Thus (31) implies (28) holds near $b = -1$, and conversely, (28) implies that r_0, c, and d

may be taken to satisfy (31) and hence (24). In particular, (31) and (25) are also consistent with each other.

So to sum up, there are continuously differentiable functions $r_0(b)$, $c(b)$, and $d(b)$ (and consequently $a(b) = (1-b)r_0(b)$) satisfying (25), (28), and $0 < d'(-1) < 1$. We have established that all of this suffices to be able to choose, given any n, values of x_0 and b which result in the IFS (f_0, f_1, x_0) witnessing the strings $0^{2n}1(01)^m$ for all $m \geq 0$. This finishes the proof of the final subcase of Theorem 4.3, and at last the proof of Theorem 4.1 is complete. □

4.4. **Further remarks.** The proof of Theorem 4.3 appears to explicitly rely on the use of IFSs over a two-letter alphabet, and a priori does not extend to show that, e.g., $A_P(0^n1^n) = 2$ may be witnessed by an IFS over $\{0, 1, 2\}$, for which another map f_2 must be specified. However, if one defines $f_0x = a + bx$ and $f_1x = c + dx$ as in any of the proofs in the last section, and lets $f_jx = \frac{a+c}{2} + \frac{b+d}{2}x$ for all other $j \in \Sigma$, then f_jx is strictly between f_0x and f_1x except at $x = i_x$, and so a string containing a j can have neither minimal nor maximal probability. Hence $A_P(w) = 2$ over a two-letter Σ implies that $A_P(w) = 2$ over any $\Sigma' \supset \Sigma$.

Theorem 4.1 immediately implies that the set of binary strings with $A_P = 2$ is a regular language. More particularly, the proof of Theorem 4.2 has the following consequence, which is somewhat intriguing given that stochastic languages—which are defined by fixed probability thresholds (the cut-point)—are not generally regular, or even recursively enumerable, although Rabin did show that a stochastic language defined by an isolated cut-point is regular [21].

Corollary 4.12. *For every two-state PFA M over a binary alphabet, the language of strings whose complexity is witnessed by M is regular.*

Proof. By the proof of Theorem 4.2, given M, the set of strings W witnessed by M consists of one of the following plus at most finitely many other strings:

- nothing,
- 0^n for all n,
- 0^n1^m for some n and all m,
- $0^n(10)^m$ for some n and all m,
- $0^n1(01)^m$ for some n and all m,
- $1^n(01)^m$ for some n and all m,
- $1^n0(10)^m$ for some n and all m,
- 1^n0^m1 for some n and all m, or
- the set of bit-flips of any one of the above.

In all cases, for all but finitely many w, we have $w \in W$ if and only if w begins with a fixed prefix and ends with a repeated pattern of length 1 or 2, possibly followed by a single extra digit. Each case can be described by a regular expression. □

Another consequence of the classification is that we can save an arbitrarily high number of states by switching from NFAs to PFAs to describe a given string:

Corollary 4.13. *$A_N(w) - A_P(w)$ may be arbitrarily large among binary w.*

Proof. The statement follows if we can show $A_N(0^n1^n)$ is unbounded in n,[3] since $A_P(0^n1^n) = 2$ for all n by Theorem 4.1. By the pigeonhole principle, it suffices to

[3] [23, Theorem 12] establishes that $A_D(0^n1^n) \geq \sqrt{n} - 1$ for all n, but the proof does not quite go through for NFAs. Probably a similar explicit lower bound on A_N can be found.

show that no NFA M can witness $A_N(0^n1^n)$ for more than finitely many values of n. If the digraph of M has fewer than two distinct cycles leading to an accepting state, then at most one string of the form 0^n1^n is accepted. Otherwise, suppose there are distinct cycles of lengths a and b, respectively. Then for any string w accepted by M, the portion of w which was read while traversing these cycles has length $\ell = ax + by$ for some $x, y \in \mathbb{N}$. But if such an ℓ is greater than $2ab - a - b$, then there are at least two different pairs of natural numbers (x, y) and (x', y') with $ax + by = ax' + by' = \ell$ (see, e.g., [23, Lemma 11]). In terms of M, this means for all large enough m such that M accepts a word of length m, there are at least two distinct accepting paths of that length, corresponding to traversing the two cycles x and y times on the one hand, and x' and y' times on the other hand. Thus M cannot witness $A_N(w)$ if $|w|$ is sufficiently large, and we are done. □

Of course, the 2-state PFA describing 0^n1^n may have to be somewhat complicated, an issue we briefly return to in Section 6 below.

No evidence has yet appeared to suggest that A_P is unbounded, or even that any string has complexity greater than 3. As remarked earlier, all binary strings of length 9 or less have complexity either 2 or 3, and witnesses with three states have been found for a number of longer strings as well. Therefore, we may pose the following questions:

Question 4.14. Is A_P unbounded? If not, what is its maximum value? Similarly when restricted to a given finite alphabet.

Question 4.15. What is a tight upper bound for $A_P(w)$ as a function of $|w|$?

Lastly, one may call a string *random* for a measure of complexity if its complexity is the maximum possible for its length. For example, for Kolmogorov complexity, a string is random if its complexity is equal to its length, up to an additive constant not depending on the string. For A_N, the string w is random if $A_N(w) = \lfloor |w| / 2 \rfloor + 1$, and this is known to be tight. But without a general asymptotic upper bound, it is unclear what strings could be considered random for A_P, and so we ask:

Question 4.16. Is there a suitable notion of a string being random with respect to PFA complexity? If so, how many strings are PFA-random (asymptotically)?

5. Computability of $A_{P,\delta}$

One of the primary motivations for the introduction of A_D was its computability, in contrast with Kolmogorov complexity. Computability would of course be a desirable property of A_P as well, but this is currently not known. One can at least show that the relation $A_P(w) \leq n$ is c.e.: any witness for $A_P(w) \leq n$ can be arbitrarily well approximated by a rational witness, since $\rho_A(w)$ is continuous in the entries of A, and to continue witnessing $\text{gap}(w) > 0$ one only needs to make sure that the probabilities of the finitely many other strings of length $|w|$ do not stray too far. Rational PFAs can be computably enumerated, clearly. When representing rationals as pairs of natural numbers, the exact value of the acceptance probability of any word can be computed in finite time, and the relations $<$, $>$, and $=$ are all decidable for any numbers resulting from such a computation. Therefore, if a witness for $A_P(w) \leq n$ exists, it will be found in finite time. Whether $A_P(w) \geq n$ is also a c.e. relation remains open.

On the other hand, the question of the computability of $A_{P,\delta}(w)$ is completely settled apart from the case $\delta = 0$:

Theorem 5.1. *For any finite alphabet Σ, the function $(\delta, w) \mapsto A_{P,\delta}(w)$ is*

- *Continuous everywhere on $[0,1) \times \Sigma^*$ except on a countably infinite set which can be enumerated by a single algorithm;*
- *Computable on $(0,1) \times \Sigma^*$ where it is continuous.*

In particular, for every w, $A_{P,\delta}(w)$ is computable for all but at most $A_D(w) - 2$ many values of δ, and is continuous at $\delta = 0$.

The proof of this theorem uses some machinery from computable analysis, and we introduce the needed background in the next subsection. For reasons that will become apparent after the proof is complete, it does not extend in any obvious way to a proof that A_P is computable. Indeed, A_P may well not be computable, but it is still not clear what tools one might use to show that, as the usual proof of the noncomputability of Kolmogorov complexity by a version of Berry's paradox is not obviously adaptable to PFAs. Alternatively, one could view the calculation of $A_P(w)$ as the decision problem which asks, given w and k, whether there is a k-state PFA M such that $\mathrm{gap}_M(w) > 0$. The hope would then be to find a reduction to this problem from a problem known to be undecidable. Many such decision problems have been studied in the theory of PFAs; the interested reader may consult the survey [10] for some important examples.

5.1. Preliminaries from computable analysis.

We follow the approach of [9]. For a separable metric space (X, d), suppose we are given an enumeration $\alpha \colon \mathbb{N} \to X$ of a dense subset of X. Fix some enumeration $(q_i)_{i \in \mathbb{N}}$ of $\mathbb{Q}$. Then we say X is a *computable metric space* if $d \colon X \times X \to \mathbb{R}$ is computable when restricted to the range of α, in the sense that the set

$$\{\, (i,j,n,m) \in \mathbb{N}^4 : q_i < d(\alpha(n), \alpha(m)) < q_j \,\}$$

is c.e. The function α gives rise to a canonical computable enumeration of a basis for the topology on X, namely

$$\langle i,j \rangle \mapsto B_{q_j}(\alpha(i)),$$

where $B_q(x)$ is the open ball of radius q centered at $x \in X$. We will from now on refer to the sets in this canonical enumeration as *basic open balls.* We may refer to a procedure as "outputting an open ball" or "listing open balls" when we really mean that it produces an index $\langle i,j \rangle$ for a basic open ball, or a list of such indices.

A *name* for a point $x \in X$ is a list N_x^X (in any order) of all basic open balls in X containing x. If (X, d_X) and (Y, d_Y) are two computable metric spaces, a function $f \colon X \to Y$ is *computable* if there is a Turing functional which sends N_x^X to $N_{f(x)}^Y$ for all $x \in X$. A *Cauchy name* for a point x is a sequence $(x_n) \subset D$ converging to x such that for all n, $d(x_n, x_{n+1}) < 2^{-n}$. One can compute a Cauchy name for x from N_x^X by first finding a subsequence of basic open balls listed in N_x^X with exponentially decreasing radii, then taking their centers. Conversely, one can compute a name N_x^X from a Cauchy name: if (x_n) is a Cauchy name for x and $B_q(y)$ is any basic open ball, then $d(x,y) < q$ iff $d(x_n, y) < q - 2^{-n}$ for some n, and the latter will be witnessed in finite time since by assumption $d(x_n, y)$ is computable in the sense given above. Neither algorithm depends on x, and so if f is computable in the above sense, then there is also a uniform computable procedure mapping a Cauchy name for x to a Cauchy name for $f(x)$ for all x. Every computable function is continuous.

The real line $\mathbb{R}$ is a computable metric space with the usual Euclidean metric, taking $D = \mathbb{Q}$. A computable real number is a number having a computable Cauchy name, viewed as an element of Baire space. If $f, g\colon X \to \mathbb{R}$ are computable functions, then so are $f + g$, $f - g$, fg, $\max\{f, g\}$, and $\min\{f, g\}$. In particular, by taking both f and g to be the identity map on $\mathbb{R}$, we get that the function $(x, y) \mapsto \max\{x, y\}$ is computable. If given $x \neq y$, one can also decide in finite time from their Cauchy names which is larger.

A computable metric space X is *computably compact* if there is a computable function which outputs a finite open cover of X by basic open balls of radius at most 2^{-n}, given input n. If $f\colon X \to \mathbb{R}$ is computable and X is computably compact, then $\sup_{x\in X} f(x)$ and $\inf_{x\in X} f(x)$ are computable numbers, and this is uniform in f (identifying f with an index for an oracle Turing machine mapping $x \mapsto f(x)$).

5.2. **Proof of Theorem** 5.1. For any $k \geq 2$, let $\mathscr{A}_k$ denote the space of k-state PFAs over a fixed finite alphabet Σ, where we identify Σ with $\{0, \ldots, b - 1\}$. To be precise, define

$$\mathscr{A}_k = \Big\{(\vec{\pi}, P_0, P_1, \ldots, P_{b-1}, \vec{\eta}) : \vec{\pi} \in [0,1]^k \text{ is a probability vector,}$$
$$\text{each } P_\sigma \text{ is a } k \times k \text{ stochastic matrix, and } \vec{\eta} \in \{0,1\}^k\Big\} \subset [0,1]^{2k+bk^2}.$$

If $A \in \mathscr{A}_k$, write the components of A as $\vec{\pi}^A, P_0^A, \ldots, P_{b-1}^A$, and $\vec{\eta}^A$. Also write M^A for the vector $(\vec{\pi}^A, P_0^A, \ldots, P_{b-1}^A)$. We give $\mathscr{A}_k$ the uniform (maximum) distance $d(\cdot,\cdot)$, i.e., that induced from the product topology on $[0,1]^{2k+bk^2}$. (The euclidean distance would work just as well.) Then $\mathscr{A}_k$ is a computably compact metric space. There are several easy ways to see this, but we give a direct proof for convenience. Let Q_k be the set of rational k-state PFAs, that is, the set of $A \in \mathscr{A}_k$ such that all entries of M^A are rational, given as quotients of natural numbers. Clearly Q_k has a computable enumeration and is dense in $\mathscr{A}_k$, and $d(A, B)$ is computable for any $A, B \in Q_k$, hence $\mathscr{A}_k$ is a computable metric space. Then for any fixed n, one can enumerate all $A \in Q_k$ such that every entry of M^A is equal to $j2^{-n-1}$ for some $j \in \{0, \ldots, 2^{n+1}\}$. The set of $B_{2^{-n}}(A)$ for all such A is a finite open cover of $\mathscr{A}_k$ by basic open balls of radius at most 2^{-n}, so $\mathscr{A}_k$ is computably compact by definition.

Recall that for any PFA A and $w \in \Sigma^*$, we have defined

$$\operatorname{gap}_A(w) = \min\{\, \rho_A(w) - \rho_A(z) : z \in \Sigma^{|w|} \setminus \{w\}\,\}.$$

The function $(A, w) \mapsto \rho_A(w)$ is computable, because it is a polynomial in the entries of A resulting from multiplication of $\vec{\pi}^A$, $\vec{\eta}^A$, and the matrices P_σ^A in an order determined by w. Therefore $(A, w) \mapsto \operatorname{gap}_A(w)$ is the minimum of finitely many computable functions and hence itself computable, as is $A \mapsto \operatorname{gap}_A(w)$ for any fixed w.

Define $\gamma^k(w) = \max_{A\in\mathscr{A}_k} \operatorname{gap}_A(w)$. Then for each k and w, $\gamma^k(w)$ is a computable real number, because it is equal to the supremum of the computable function $A \mapsto \operatorname{gap}_A(w)$ over the computably compact space $\mathscr{A}_k$. And since the procedure to compute $\operatorname{gap}_A(w)$ is uniform in w, the function $(k, w) \mapsto \gamma^k(w)$ is computable. Finally, let

$$E = \{\, (\gamma^k(w), w) : 2 \le k \le A_D(w) - 1,\ w \in \Sigma^*, 0 < \gamma^k(w) < 1 \,\} \subset (0,1) \times \Sigma^*.$$

This will turn out to be exactly the set of discontinuities of $A_{P,\delta}(w)$, and it can clearly be enumerated by a single algorithm by definition. Proposition 3.1(ii) implies that $A_{P,\delta}(w)$ is continuous at $(0, w)$ for all w. Continuity on the remainder of the complement of E will follow from the computability argument below.

That E is countably infinite is a consequence of the following fact of potential independent interest, whose proof establishes that in some sense, a 2-state PFA giving a gap of 1 to even a single word (with more than three letters) behaves much like a DFA as far as A_P is concerned.

Lemma 5.2. *For any w with $|w| \ge 4$, $\gamma^2(w) = 1$ iff w is constant.*

Proof. The right-to-left implication is immediate, since then $A_D(w) = 2$. For the other direction, assume for sake of contradiction that w is nonconstant, and that $\gamma^2(w) = 1$ is witnessed by the IFS with starting value x_0 and maps $f_j x = a_j + b_j x$ for each letter $j \in \Sigma$. Then $\rho(w) = 1$ and $\rho(z) = 0$ for every other z of length $|w|$, and if $w = z^\frown i$ where $i \in \Sigma$, then in particular $f_i \rho(z) = 1$ and $f_j \rho(z) = 0$ for all $j \ne i$. Now, if the range of f_i omits the value 0, then $\rho(i^n) > 0$ for all n, regardless of the value of x_0. Then either w is constant or $\mathrm{gap}(w) < 1$, a contradiction, and we may thus assume the range of f_i includes both 0 and 1. By drawing a picture, one sees that only $f_i x = x$ and $f_i x = 1 - x$ are possible. If $f_i x$ is the identity then only constant strings may be witnessed, so we can assume that $f_i x = 1 - x$.

If $f_i x = 1 - x$, then $\rho(z) = f_i^{-1}(1) = 0$, so $f_j 0 = 0$ and thus $f_j x = b_j x$ for all $j \ne i$. We can take $b_j < 1$, as otherwise f_j is the identity map and only constant strings can be witnessed. If $b_j = 0$ for some j, so that $f_j \equiv 0$, then every string ending in ji has probability 1, thus maximal probabilities are nonunique starting at length 3, a contradiction. Then $0 < b_j < 1$ for all $j \ne i$, and this means once an orbit leaves $\{0, 1\}$ it can never return to either value. In particular, $x_0 \in \{0, 1\}$. If $x_0 = 0$, then for all $n \ge 1$ we have $\rho(ji^{2n-1}) = \rho((ji)^{2n}) = 1$ among even-length strings and $\rho(i^{2n+1}) = \rho(j^2 i^{2n-1}) = 1$ among odd-length strings. If $x_0 = 1$, then for all $n \ge 1$ we have $\rho(i^{2n}) = \rho(ij^{2n-2}i) = 1$ among even-length strings and $\rho(ij^{2n-1}i) = \rho(iji^{2n-1}) = 1$ among odd-length strings. Either way, uniqueness of maxima is lost starting at length at most 4, so $\mathrm{gap}(w) < 1$ and by contradiction the proof is complete. □

There are infinitely many nonconstant w with $|w| \ge 4$ and $A_P(w) = 2$, of course, by Theorem 4.1. For such w, the lemma implies that $0 < \gamma^2(w) < 1$, so that $(\gamma^2(w), w) \in E$ and in particular E is infinite.

We now show that $A_{P,\delta}(w)$ is discontinuous on E and computable on the complement of E, minus the points with $\delta = 0$. Endow Σ^* with the discrete topology in its standard metrization, i.e., $d(x, y) = 1$ iff $x \ne y$. Then we give $[0, 1) \times \Sigma^*$ the product metric, that is, $d((\alpha, x), (\beta, y)) = \max\{|\alpha - \beta|, d_{\Sigma^*}(x, y)\}$. The codomain $\mathbb{N}$ of $A_{P,\delta}(w)$ also has the discrete topology as a subset of $\mathbb{R}$. Now, $A_{P,\delta}(w)$ is continuous at (δ, w) iff for all $\varepsilon > 0$ there is an $\eta > 0$ such that $d((\delta, w) - (\delta', w')) < \eta$ implies $|A_{P,\delta'}(w) - A_{P,\delta}(w)| < \varepsilon$—so that actually $|\delta - \delta'| < \eta$ implies $A_{P,\delta'}(w) = A_{P,\delta}(w)$ (since Σ^* and $\mathbb{N}$ both have the discrete topology). If $\delta = \gamma^k(w)$ for some k and w, then by definition of γ^k there is no $\delta' < \delta$ such that $A_{P,\delta'}(w) = A_{P,\delta}(w)$, because there is a k-state PFA having a gap greater than δ' for w but not one having a gap greater than δ. Hence $A_{P,\delta}(w)$ is discontinuous at every point of E.

Finally, let $(\delta, w) \notin E$ be given with $\delta \neq 0$. Under these hypotheses, for any $k \geq 2$, we have $\delta > \gamma^k(w)$ if and only if $A_{P,\delta}(w) > k$, because in this case there is no $A \in \mathscr{A}_k$ exhibiting the required gap. Conversely, $\delta < \gamma^k(w)$ if and only if $A_{P,\delta}(w) \leq k$. To compute $A_{P,\delta}(w)$, then, decide for each $k = 2, 3, \ldots, A_D(w)$ whether δ or $\gamma^k(w)$ is greater. $A_{P,\delta}(w)$ is equal to the least k such that $\delta < \gamma^k(w)$. It is clear this procedure does not depend on δ or w, and the proof of Theorem 5.1 is complete.

5.3. **Remarks.** It is worth drawing attention to the fact that if one picks any $A_D(w) - 1$ distinct values of δ, then the computation of $A_{P,\delta}(w)$ is guaranteed to converge for at least one of these δs. However, this does not allow one to extend the above argument to show that $A_P = A_{P,0}$ is computable, because $\gamma^k(w) = 0$ for all $k < A_P(w)$ and the argument does not work if $\delta = \gamma^k(w)$. (In other words, $\delta = 0$ will never be a value for which the algorithm we have given can work, unless $A_P(w) = 2$.) One would have to find the least k such that $\gamma^k(w) > 0$ and compute $A_{P,\delta}(w)$ for any $0 < \delta < \gamma^k(w)$ (for that k). In general, identifying the least positive element of a finite set of computable real numbers is undecidable.

The function $(k, w) \mapsto \gamma^k(w)$ seems to be of interest in and of itself, as we briefly discuss in the following section. We have $0 \leq \gamma^2(w) \leq \gamma^3(w) \leq \cdots \leq \gamma^{A_D(w)}(w) = 1$, with $\gamma^k(w) = 0$ if and only if $k < A_P(w)$. $\gamma^k(w)$ is never negative since one can always make a PFA accepting every word with the same probability by setting all transition matrices to the identity matrix. Furthermore, Proposition 3.3 implies $\gamma^k(z) \geq \gamma^k(wz)$ for all w, z, and k. This comes close to justifying the empirical observation made in Section 3 that gaps tend to decrease for longer words. A result to the effect that $\gamma^k(w) \geq \gamma^k(wz)$ would put the observation on fully rigorous ground.

6. Other approaches to probabilistic complexity

6.1. **Relaxing the definition of a PFA.** We saw earlier that A_P shares the property of A_D that the complexity of a string is not necessarily equal to that of its reversal. In addition, as noted in the introduction, there are strings whose PFA complexity is known to be witnessed by a PFA with dead states. One might try to rectify these issues by relaxing the definition of a PFA to directly generalize an NFA (rather than a DFA). One way to do so is prompted by the observation that an NFA is allowed to have rows of all zeros in its transition matrices. NFAs also have the property that different out-transitions from the same state and for the same letter are not weighted differently—they are simply all possible. The same applies to the initial states.

To directly translate these properties to a relaxed version of a PFA, one would need to require that all nonzero entries of $\vec{\pi}$ are equal, and that all nonzero entries of all the matrices P_σ are equal to the same number (which may result in the row sums being different). One can see that the proof of the first part of Proposition 3.4 can be recovered for the class of such automata, so that if $\tilde{A}_P$ is the corresponding complexity notion then $\tilde{A}_P(\overleftarrow{x}) = \tilde{A}_P(x)$ for all x (and moreover $\tilde{A}_P(x) \leq A_N(x)$). However, this is not a very natural class of automata to consider and it is certainly not a direct generalization of a PFA.

Instead of trying to design a specific class of automata in an attempt to recover properties of A_N, it might make more sense to define a unified complexity notion which takes as parameter a family of automata and study its properties in general. In [25], Turakainen introduced *generalized (probabilistic) finite automata* (GPFAs), which are finite-state automata whose operation is described as follows:

- The initial state of the machine is an arbitrary real row vector.
- Transitions between states are described by multiplication of arbitrary real square matrices.
- The final state of the machine is again an arbitrary real column vector.

So GPFAs are like PFAs except that the entries of $\vec{\pi}$, $\vec{\eta}$, and each P_σ can be any real numbers. Turakainen proved the remarkable fact that GPFAs have in a sense the same descriptive power as PFAs: if one also allows a cut-point in the context of a GPFA to be any real number, then the class of languages accepted by GPFAs is exactly the class of stochastic languages.

This suggests that it is not too unreasonable to throw the gates open and consider a version of A_P that allows any GPFA. Let $\mathscr{G}$ be the set of all GPFAs. For any family $\mathscr{F} \subseteq \mathscr{G}$, let $\mathscr{F}_k$ be the set of members of $\mathscr{F}$ having k states. Then define the *$\mathscr{F}$-automatic complexity* of a word $x \in \Sigma^*$ to be

$$A_{\mathscr{F}}(x) = \min\{\, k : \exists F \in \mathscr{F}_k \text{ such that } \operatorname{gap}_F(x) > 0 \,\}.$$

For example, A_P as defined before coincides with $A_{\mathscr{P}}$ if $\mathscr{P} \subset \mathscr{G}$ is the set of all PFAs. One can also define $A_{\mathscr{F},\delta}$ for any $\delta \geq 0$ by analogy with $A_{P,\delta}$. We have that for all x,

$$A_{\mathscr{E}}(x) \leq A_{\mathscr{F}}(x) \quad \text{whenever} \quad \mathscr{E} \supseteq \mathscr{F},$$

so that in particular $A_{\mathscr{G}}(x) \leq A_{\mathscr{F}}(x)$ for every $\mathscr{F}$ and x. We have not investigated $A_{\mathscr{F}}$ in general, and it is unclear how coarse of a measurement it might be. As a motivating question, we could ask

Question 6.1. Is $A_{\mathscr{G}}(x) \leq 2$ for all binary strings x?

It is at least true that $A_{\mathscr{G},\delta}(x) = A_{\mathscr{G},\delta}(\overleftarrow{x})$ for all x and $\delta \geq 0$, since the proof of Proposition 3.4 works for any GPFA—one can simply make $\vec{\pi}' = \vec{\eta}$, $\vec{\eta}' = \vec{\pi}$, and $P'_\sigma = P^T_\sigma$ with no rescaling needed. Together with Proposition 3.3, whose proof goes through verbatim for $A_{\mathscr{G},\delta}$, this implies that $A_{\mathscr{G},\delta}(xyz) \geq A_{\mathscr{G},\delta}(y)$ for all x, y, and z, like A_D and A_N.

A potentially helpful observation here is that the ability to have unbounded real entries does not really confer any advantage as far as the complexity of individual strings is concerned. For any GPFA M, if C is the largest absolute value of any entry of $\vec{\pi}^M$, $\vec{\eta}^M$, and the matrices P^M_σ, then one could divide all these matrices and vectors by C to obtain a GPFA M' with entries in $[-1, 1]$ such that

$$\rho_M(x) < \rho_M(y) \iff \rho_{M'}(x) < \rho_{M'}(y)$$

whenever $|x| = |y|$. Hence if $\mathscr{S}$ is the set of GPFAs whose entries are all in $[-1, 1]$, we have $A_{\mathscr{S}}(x) = A_{\mathscr{G}}(x)$ for all x. In addition, the direct analogue of Theorem 5.1 holds for $A_{\mathscr{S},\delta}$, because $\mathscr{S}_k$ is now a computably compact metric space for each k, unlike $\mathscr{G}_k$.

One advantage of A_P that appears to be immediately lost in passing to $A_{\mathscr{G}}$ or $A_{\mathscr{S}}$ is the dimension reduction of the IFS approach, and the dynamical analysis made more tractable by it. Since the correspondence between PFAs and IFSs relies explicitly on the transition matrices being stochastic, perhaps one could allow only generalized stochastic transition matrices, with any real entries permitted as long as each row sums to 1. This notion would for example allow us to describe 0100 in two states via

$$P_0 = \begin{pmatrix} -1 & 2 \\ 1/2 & 1/2 \end{pmatrix}, \quad P_1 = \begin{pmatrix} 1/2 & 1/2 \\ 1 & 0 \end{pmatrix}, \quad \vec{\pi} = (0,1), \quad \vec{\eta} = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$

whereas $A_P(0100) = 3$, so strictly greater compression is achieved. This automaton is equivalent to the IFS with $f_0(x) = \frac{1}{2} - \frac{3}{2}x$, $f_1(x) = 1 - \frac{1}{2}x$, and $x_0 = 0$. (Other strings with $A_P = 3$ which have complexity 2 according to this notion include 01000, 01011, and 01100.) Unfortunately, uniformly rescaling the transition matrices as with $\mathscr{S}$ no longer works here, so the set of allowed transition probabilities is unbounded and we lose computability of the analogue of $A_{P,\delta}$, i.e., the proof of Theorem 5.1 cannot be recovered.

6.2. **Gap structure function.** We saw in the proof of Theorem 5.1 that the function $\gamma^k(w)$ mapping w to the maximum value of $\text{gap}_M(w)$ among all k-state M is computable. It could be interesting to study $\gamma^k(w)$ as a parametrized complexity measure in itself. Intuitively, $w \mapsto \gamma^k(w)$ measures how well w is described by the model class of k-state PFAs—the widest margin of probability by which w can be recognized by any such PFA. This relates $\gamma^k(w)$ at least philosophically to the Kolmogorov structure function, which measures the minimal size of a set of strings containing w which can be described by a Turing machine of size at most k, and hence captures in a sense how well w can be singled out by such machines. Similar functions have also been considered by Kjos-Hanssen [16], who introduced both a structure function and a dual structure function for the NFA complexity. His dual structure function has the desirable feature, he points out, of a simple domain and complicated range, rather than the other way around as for his regular structure function for A_N. This is even more true for $\gamma^k(w)$ in contrast with its dual $A_{P,\delta}(w)$, especially if one is interested in computability.

6.3. **Least number of bits of a witness.** Heuristically it appears that witnesses for the PFA complexity of many strings are relatively complicated; this certainly seems to be the case for most strings with $A_P = 2$, as pointed out below. If one is interested solely in compression, it might make the most sense to measure the complexity of w as the least number of bits required to describe an M having $\text{gap}_M(w) > 0$, or perhaps $\text{gap}_M(w) > \delta$ for a parameter δ. One potential drawback of this approach is that it is not obvious whether this measure is computable, although this depends on the precise definition used. The least number of bits also depends on the choice of encoding, and so this measure would only be defined up to an additive constant, like the Kolmogorov complexity. Not only that, but it could well be that the simplest PFAs achieving a positive gap are very often DFAs, and in that case one could argue it is hardly a satisfying notion of PFA complexity.

6.4. **Measure of the set of witnesses.** We conclude by mentioning one more idea for modifying A_P and $A_{P,\delta}$, with the aim of refining the numerical measurement itself. In the proof of for example Proposition 4.5, we saw that although all strings 0^n1^m have complexity 2, as n increases, x_0 must be chosen in a narrower and narrower range in order for the IFS to witness 0^n1^m. The coefficient b must also be made arbitrarily close (but not equal) to 1. Something similar is true of the other subcases of the proof of Theorem 4.3. Thus it is in a sense more complicated to witness the complexity of a string the longer its prefix is. So, we could introduce a real-valued complexity measure that accounts for that difference as follows. Let μ be a Borel probability measure with full support on $\mathscr{A}_k$, the space of k-state PFAs. Let $G^k(x) = \mathrm{gap}_\bullet(x)^{-1}((0,1])$ be the set of k-state witnesses for $A_P(x) \leq k$, and let

$$A_\mu(x) = A_P(x) + 1 - \mu(G^k(x)).$$

We can also let $G^k_\delta(x) = \mathrm{gap}_\bullet(x)^{-1}((\delta,1])$ and define

$$A_{\mu,\delta}(x) = A_{P,\delta}(x) + 1 - \mu(G^k_\delta(x)).$$

Since gap is a computable function on a computably compact metric space, $G^k(x)$ and $G^k_\delta(x)$ are c.e. open, meaning the indices of all basic open balls contained in each of them can be computably enumerated. In particular, all these sets have positive μ-measure if nonempty. Thus $A_\mu(x)$ assigns x a value strictly between $A_P(x)$ and $A_P(x) + 1$, and $A_{\mu,\delta}(x)$ is strictly between $A_{P,\delta}(x)$ and $A_{P,\delta}(x) + 1$. Moreover, in at least the case of binary strings with $A_P(x) = 2$, if x is a string such as 0^n101 which can only be witnessed by a PFA that also witnesses $0^n1(01)^m$ for all m, then those strings receive exactly the same value of A_μ as x does. This makes sense, because these extensions of x in a sense do not require any further effort to find a witness. The latter observation holds for any measure μ. The goal in defining A_μ (or $A_{\mu,\delta}$) would then be to find a suitable μ which gives sets like $G^k(x)$ (or $G^k_\delta(x)$) large measure for strings like $x = (01)^n$ which are easy to witness, while giving smaller measure to $G^k(x)$ (or $G^k_\delta(x)$) for strings whose witnessing automata require a more precise configuration.

As with most of our proposed notions of probabilistic complexity, it is hardly clear from the definition if $A_{\mu,\delta}$ is computable, let alone A_μ, even if the measure μ is required to be computable and if $\delta \notin E$ (where E is as in Section 5.2). We close by asking

Question 6.2. If μ is a computable Borel probability measure on $\mathscr{A}_k$ with full support, is $(\delta, x) \mapsto A_{\mu,\delta}(x)$ necessarily computable on $((0,1) \setminus E) \times A_P^{-1}(k)$? If not, what would be a natural choice of μ to make it computable?

Question 6.3. How should the definitions of A_μ and $A_{\mu,\delta}$ account for the fact that lower-complexity strings are also witnessed by members of $\mathscr{A}_k$? How should one deal with the likely problem of the sets $G^k(x)$ generally having high measure when $A_P(x) < k$, which would make A_μ clustered near $k+1$ among strings having $A_P(x) = k$?

References

[1] Hideo Bannai et al. "The smallest grammar problem revisited". In: *IEEE Trans. Inf. Theory* 67 (1 2021), pp. 317–328. arXiv: 1908.06428.

[2] Michael F. Barnsley and Lyman P. Hurd. *Fractal image compression.* AK Peters, 1993.

[3] Cristian S. Calude, Kai Salomaa, and Tania K. Roblot. "Finite state complexity". In: *Theoret. Comput. Sci.* 412 (2011), pp. 5668–5677. DOI: 10.1016/j.tcs.2011.06.021.

[4] J. W. Carlyle. “Reduced forms for stochastic sequential machines”. In: *J. Math. Anal. Appl.* 7 (1963), pp. 167–175.

[5] Rohit Chadha, A. Prasad Sistla, and Mahesh Viswanathan. “Probabilistic automata with isolated cut-points”. In: *MFCS 2013 (LNCS, vol. 8087).* Ed. by K. Chatterjee and J. Sgall. Springer, 2013, pp. 254–265.

[6] Karel Culik II and Simant Dube. “Affine automata and related techniques for generation of complex images”. In: *Theoret. Comput. Sci.* 116 (1993), pp. 373–398.

[7] Karel Culik II and Simant Dube. “Rational and affine expressions for image description”. In: *Discrete Appl. Math.* 41 (1993), pp. 85–120.

[8] A. A. Diwan. *A new combinatorial complexity measure for languages.* Tech. rep. Computer Science Group, Tata Institute, 1986.

[9] Rodney G. Downey and Alexander G. Melnikov. *Computably compact metric spaces.* URL: `https://homepages.ecs.vuw.ac.nz/~downey/publications/compcomp.pdf`.

[10] Nathanaël Fijalkow. “Undecidability results for probabilistic automata”. In: *ACM SIGLOG News* 4.4 (Oct. 2017). DOI: `10.1145/3157831.3157833`.

[11] Kenneth Gill. “Two studies in complexity”. PhD thesis. Penn State University, 2023.

[12] Kayleigh Hyde. “Nondeterministic finite state complexity”. MA thesis. University of Hawai’i, Manoa, 2013.

[13] Kayleigh Hyde and Bjørn Kjos-Hanssen. “Nondeterministic automatic complexity of overlap-free and almost square-free words”. In: (2020). arXiv: `1402.3856`. An earlier version of the paper was published in *Electronic J. Comb.* in 2015 as paper 3.22.

[14] Bjørn Kjos-Hanssen. “An incompressibility theorem for automatic complexity”. In: *Forum of Mathematics, Sigma* 9 (2021). Paper e62. DOI: `10.1017/fms.2021.58`. arXiv: `1908.10843`.

[15] Bjørn Kjos-Hanssen. *Automatic complexity: A computable measure of irregularity.* De Gruyter, 2024. DOI: `10.1515/9783110774870`.

[16] Bjørn Kjos-Hanssen. “Kolmogorov structure functions for automatic complexity”. In: *Theoret. Comput. Sci.* 607 (2015), pp. 435–445. DOI: `10.1016/j.tcs.2015.05.052`. arXiv: `1409.0584`.

[17] Bjørn Kjos-Hanssen. “Maximal automatic complexity and context-free languages”. In: *Aspects of Computation and Automata Theory with Applications.* 2023, pp. 335–352. DOI: `10.1142/9789811278631_0013`. arXiv: `2206.10130`.

[18] Bjørn Kjos-Hanssen. “On the complexity of automatic complexity”. In: *Theory Comput. Syst.* 61 (2017), pp. 1427–1439. arXiv: `1607.06106`.

[19] Ljubiša M. Kocić and Alba C. Simoncelli. “Cantor dust by AIFS”. In: *Filomat* 15 (2001), pp. 265–276. URL: `http://www.jstor.org/stable/26453430`.

[20] Azaria Paz. *Introduction to probabilistic automata.* New York: Academic Press, 1971.

[21] Michael O. Rabin. “Probabilistic automata”. In: *Inform. and Control* 6 (1963), pp. 230–245.

[22] I. K. Rystsov. “Affine automata and classical fractals”. In: *Cybernet. Systems Anal.* 54.1 (Jan. 2018). DOI: `10.1007/s10559-018-0003-6`.

[23] Jeffry Shallit and Ming-wei Wang. “Automatic complexity of strings”. In: *J. Autom. Lang. Comb.* 6.4 (2001), pp. 537–554.

[24] J. C. Sprott. “Automatic generation of iterated function systems”. In: *Comput. & Graphics* 18.3 (1994), pp. 417–425.

[25] Paavo Turakainen. “Generalized automata and stochastic languages”. In: *Proc. Amer. Math. Soc.* 21 (1969), pp. 303–309.

[26] Enrique Vidal et al. “Probabilistic finite-state machines—Part I & II”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.7 (2005), pp. 1013–1039. DOI: 10.1109/TPAMI.2005.147.

Email address: gillmathpsu@posteo.net