Calibrating Generative Models: The Probabilistic Chomsky-Schützenberger Hierarchy*

Thomas F. Icard
Stanford University
icard@stanford.edu

December 14, 2019

1 Introduction and Motivation

Probabilistic computation enjoys a central place in psychological modeling and theory, ranging in application from perception (Knill and Pouget, 2004; Orbán et al., 2016), attention (Vul et al., 2009), and decision making (Busemeyer and Diederich, 2002; Wilson et al., 2014) to the analysis of memory (Ratcliff, 1978; Abbott et al., 2015), categorization (Nosofsky and Palmeri, 1997; Sanborn et al., 2010), causal inference (Denison et al., 2013), and various aspects of language processing (Chater and Manning, 2006). Indeed, much of cognition and behavior is usefully modeled as stochastic, whether this is ultimately due to indeterminacy at the neural level or at some higher level (see, e.g., Glimcher 2005). Random behavior may even enhance an agent's cognitive efficiency (see, e.g., Icard 2019).

A probabilistic process can be described in two ways. On one hand we can specify a stochastic procedure, which may only implicitly define a distribution on possible outputs of the process. Familiar examples include well known classes of generative models such as finite Markov processes and probabilistic automata (Miller, 1952; Paz, 1971), Boltzmann machines (Hinton and Sejnowski, 1983), Bayesian networks (Pearl, 1988), topic models (Griffiths et al., 2007), and probabilistic programming languages (Tenenbaum et al., 2011; Goodman and Tenenbaum, 2016). On the other hand we can specify an analytical expression explicitly describing the distribution over outputs. Both styles of description are commonly employed, and it is natural to ask about the relationship between them. The question is one of definability or expressiveness: for a given class of generative models, what kinds of distributions (analytically characterized) can those models implicitly define?

A prominent way of classifying models of computation, and generative procedures in particular, is in terms of memory usage. In the classical non-probabilistic setting this leads to the well known Chomsky hierarchy—also called the Chomsky-Schützenberger hierarchy—of machines and grammars. At the bottom are finite-state automata (or equivalently, regular grammars), which are limited to a fixed, finite memory; at the top are Turing machines (equivalently, unrestricted grammars or "production systems"), which have unrestricted access to an unbounded memory; and in between are various models of limited-access memory such as stack automata and context-free grammars (see Figure 1). Much

*Preprint. Forthcoming in *Journal of Mathematical Psychology*.

is known about the *languages*—that is, sets of strings over a finite alphabet—definable at different levels of the hierarchy (see, e.g., Eilenberg 1974; Hopcroft and Ullman 1979).

The driving motivation for early theoretical work on formal grammars came from the psychology of language, where the focus was on finding adequate frameworks for describing and explaining human grammatical competence (Chomsky, 1959, 1965; Chomsky and Schützenberger, 1963). With increased emphasis on detailed processing, parsing, and acquisition accounts, in addition to large-scale grammar induction from corpora in computational linguistics, a considerable amount of applied work has explored probabilistic grammars (see, e.g., Klein and Manning 2003; Levy 2008; Kim et al. 2019 for representative examples), the straightforward result of adding probabilistic transitions to classical grammar formalisms. Here again, much of the interest has been to find formalisms that are powerful enough—but not too powerful—to capture relevant linguistic structure.

Aside from their prevalence in models of language processing, probabilistic grammars have featured in numerous other psychological domains. For instance, they have recently appeared in characterizations of abstract conceptual knowledge (Tenenbaum et al., 2006, 2011), in computational models of vision (Zhu and Mumford, 2007), and in accounts of how people perceive randomness (Griffiths et al., 2018), among others. However, the psychological interest of probabilistic grammars is not limited to these direct applications.

As in the classical setting, there is a systematic correspondence between probabilistic grammar classes and machine models. For example, probabilistic regular grammars can be shown expressively equivalent to hidden Markov models and probabilistic automata (see, e.g., Smith and Johnson 2007) and to classes of artificial neural networks (MacKay, 1996); probabilistic context-free grammars have been shown equivalent to branching processes (Harris, 1963), probabilistic pushdown automata (Abney et al., 1999) and so called recursive Markov chains (Etessami and Yannakakis, 2009), while also being powerful enough to define tools like topic models (Johnson, 2010); and unrestricted probabilistic grammars are equivalent to probabilistic Turing machines and indeed to any other Turing-complete probabilistic programming language (Theorem 2 below). Meanwhile, probabilistic grammars have even been invoked to calibrate and assess the capabilities of recurrent neural networks (Lin and Tegmark, 2017), and they have also been assimilated to causal models (Chater and Oaksford, 2013; Icard, 2017b). Studying the expressive capacity of probabilistic grammars thus promises to illuminate that of many other important model classes. More generally, the resulting *probabilistic Chomsky-Schützenberger hierarchy* offers a meaningful classification of discrete probabilistic models, and serves as a useful target for understanding the expressiveness of probabilistic models broadly.

As running examples, consider the following four common discrete distributions:

Example 1 (Poisson)**.** The Poisson distribution, $\mu(k) = e^{-\lambda}\frac{\lambda^k}{k!}$, is ubiquitous in psychological modeling, reaching even to the level of neural firing rates (Faisal et al., 2008). Given an average rate λ of independent occurrences in an interval, $\mu(k)$ is the probability of observing exactly k occurrences in the interval.

Example 2 (Negative Binomial)**.** Consider a measure $\mu_{q,t}(k) = \binom{t+k}{k}q^{t+1}(1-q)^k$, for $t \in \mathbb{N}$ and $q \in \mathbb{Q} \cap [0,1]$, giving the probability that we will make k observations other than our target observation by the time we see the $t+1$st target. The negative binomial distribution is often used in place of the Poisson distribution for probabilistic modeling. In fact, $\mu_{q,t}$ converges to the Poisson distribution as t goes to infinity (Johnson et al., 2005).

Example 3 (Random Walk Hitting Time)**.** Various kinds of random walks have appeared in psychological modeling (see, e.g., Townsend and Ashby 1983; Abbott et al. 2015 for very different examples). Consider the simplest symmetric 1/2-random walk on the non-negative integers, starting at 1. The hitting time for first reaching 0 is given by a distribution $\mu(2k+1) = c_k 2^{-2k+1}$, where $c_k = \binom{2k}{k}\frac{1}{k+1}$ is the kth Catalan number.

Example 4 (Beta-Binomial)**.** Hierarchical probabilistic models have generated a great deal of interest in cognitive science (e.g., Griffiths et al. 2007; Liang et al. 2010; Gershman and Blei 2012; Li et al. 2019). One of the most basic building blocks in these models is the Beta-Binomial distribution (and its multidimensional generalization, the Dirichlet-Multinomial), which defines a generative procedure for flipping coins with unknown weight. Consider a measure defined by first drawing p from a $\mathsf{Beta}(\alpha, \beta)$ distribution and then generating a number k with probability $\binom{n}{k}p^k(1-p)^{n-k}$. The distribution on k is then:

$$\mu(k) \quad = \quad \binom{n}{k}\frac{B(\alpha+k, \beta+n-k)}{B(\alpha,\beta)}$$

In fact, μ approximates a negative binomial as n and β increase (Johnson et al., 2005).

Our basic question is this: how high in the hierarchy do we need to ascend before we can (implicitly) define probability distributions like these?

The aim of the present contribution is to present a picture, as comprehensive as possible, of the expressive capacity of probabilistic grammar formalisms across the Chomsky-Schützenberger hierarchy, from probabilistic regular grammars to unrestricted grammars (Figure 1). Our study draws from a wealth of previous work in different research traditions. Most prominently, we employ analytic techniques that date back to the beginning of formal language and automata theory, to the pioneering algebraic approach of Schützenberger (Schützenberger, 1961, 1965; Chomsky and Schützenberger, 1963), which has in turn spawned a rich literature in theoretical computer science and analytic combinatorics (see Eilenberg 1974; Salomaa and Soittola 1978; Kuich and Salomaa 1986; Flajolet 1987; Flajolet and Sedgewick 2001; Droste et al. 2009, among many others). Much of this work is quite abstract, e.g., dealing with generalized notions of rational and algebraic power series. Probabilistic interpretations are often discussed as special cases (for instance, as a possible semi-ring related to a weighted grammar or automaton, see Droste et al. 2009), and some of the literature deals with languages defined by probabilistic automata with "cut-points" (Rabin, 1963; Salomaa and Soittola, 1978). However, extracting lessons for probabilistic grammars specifically often requires further work.

Meanwhile, there have been a number of important results dealing with expressivity of probabilistic grammars and machines *per se*, from mathematical psychology (e.g., Vitányi and Chater 2017), mathematical linguistics (e.g., Chi 1999; Smith and Johnson 2007; Kornai 2008), statistics (e.g., O'Cinneide 1990), and other areas of computer science (e.g., de Leeuw et al. 1956; Yao 1985). We draw on this body of work as well.

Our presentation and analysis of the probabilistic Chomsky-Schützenberger hierarchy combines a number of old and new results, and is intended to be as streamlined and self-contained as possible. Many of the observations are original, but we also aim to bring together techniques and ideas from disparate areas in order to paint an expansive picture of the hierarchy as a whole. We include proofs of all new results, as well as results whose proofs might be difficult to find or reconstruct.

After formal preliminaries (§2), we characterize the class of all probabilistic grammars showing that they define exactly the enumerable semi-measures (Theorem 3). An important question pertinent to (especially Bayesian) cognitive modeling is whether a class of distributions is closed under conditioning (a kind of counterpart to the notion of *conjugacy* in statistics). We show that distributions defined by probabilistic grammars (or machines) are not in general closed, even under conditioning with a finite set (Theorem 6). However, the probabilistic grammars that almost-surely terminate are closed under conditioning on any computable set whatsoever (Theorem 7).

Turning to the most restrictive class, the probabilistic regular grammars, we show that they are capable of defining any finite-support rational-valued distribution (Corollary 9), before giving a self-contained proof of the result that the probability generating function for a probabilistic regular grammar is always rational (Theorem 15). We also show that the distributions defined by probabilistic regular grammars are closed under conditioning with arbitrary regular sets (Theorem 17).

Probabilistic context-free grammars are shown by several examples (13-17) to define irrational generating functions; however, they are all still algebraic (Theorem 22). The result contrasts with the classical Chomsky-Schützenberger characterization (Chomsky and Schützenberger, 1963), which requires restriction to unambiguous context-free grammars (see §5.1). Perhaps surprisingly, when considering finite-support distributions—often central to applications—probabilistic context-free grammars define no more distributions than the regular grammars (Proposition 18). A consequence of this is that probabilistic context-free grammars are not closed under conditioning, even with finite sets (Proposition 19).

We consider two grammar classes that are not part of the original hierarchy, situated in between context-free and context-sensitive, namely indexed (Aho, 1968) and linear indexed (Duske and Parchmann, 1984) grammars. Both have played a notable role in computational linguistics (e.g., Gazdar 1988; Joshi et al. 1991; Kim et al. 2019). Probabilistic indexed grammars are capable of defining distributions with transcendental generating functions (Proposition 23). Even probabilistic linear indexed grammars—which possess algebraic generating functions and whose (right-linear) restrictions are weakly equivalent to context-free grammars (Aho, 1968)—can define finite-support irrational distributions, thus surpassing the power of context-free in the probabilistic setting (Proposition 24).

Finally, it is shown that probabilistic context-sensitive grammars can also define distributions with transcendental generating functions, including some that elude probabilistic indexed grammars (Propositions 27 and 28). We demonstrate that, in some sense, probabilistic context-sensitive grammars come quite close to arbitrary probabilistic grammars (Proposition 29). In other ways, however, these grammars are unnatural from a probabilistic perspective. For finite-support measures they again coincide with the regular grammars, defining only rational-valued distributions; and unlike in the classical hierarchy, they do not even extend context-free (Proposition 30).

The resulting hierarchy is summarized in Figure 1. Notably, the picture remains unchanged when restricting attention to a one-letter alphabet, a restriction that is very natural for many applications, viz. unary representations of positive integers (recall Examples 1-4). It has been observed that Parikh's Theorem (Parikh, 1966)—showing that the context-free languages are "semi-linear" and thus coextensive with the regular languages for a one-letter alphabet—does not extend to the probabilistic setting (Petre, 1999; Bhattiprolu et al., 2017). We observe that this trend is widespread, applying also to the linear indexed grammars (which are also semi-linear, see Duske et al. 1992). Another emerging theme is that, while finite languages trivialize in the classical setting, finite-support distributions

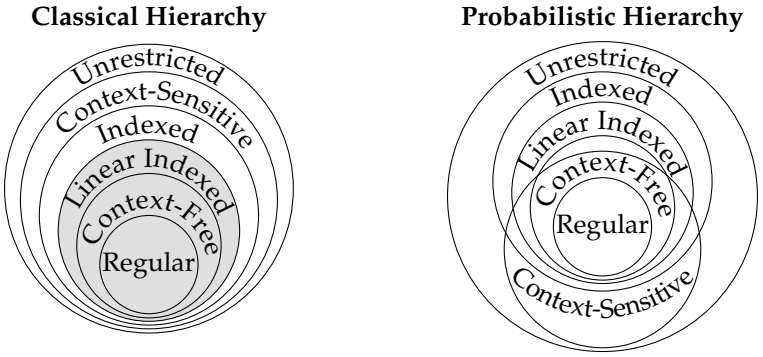

Figure 1: The classical and probabilistic hierarchies. The shaded region signifies that these classes collapse into the regular languages when considering a one-element alphabet. No such collapse occurs in the probabilistic hierarchy. Meanwhile, the probabilistic context-sensitive grammars are incomparable with the probabilistic context-free grammars.

can reveal subtle and significant distinctions in the probabilistic setting.

In the concluding section (§8) we return to consider what repercussions this pattern of results may have for psychology and cognitive science.

2 Formal Preliminaries

2.1 Strings and Distributions

Our interest is in distributions over strings from a finite alphabet Σ. We construe strings as lists, depicted by concatenation, and denote the empty string by ϵ. The length of a string $|\sigma|$ is defined so that $|\epsilon| = 0$ and $|a\sigma| = |\sigma| + 1$. Exponentiation is defined by $\sigma^0 = \epsilon$ and $\sigma^{n+1} = \sigma^n\sigma$, so that $|\sigma^n| = n|\sigma|$.

We will make use of a canonical ordering on the set of substrings of a string. Given σ and non-empty substrings σ_1, σ_2 of σ, we say σ_1 *has priority over* σ_2 if either (1) the left-most symbol of σ_1 is to the left in σ of the left-most symbol of σ_2, or (2) their leftmost symbols are identical, at the same index of σ, but σ_2 is a proper substring of σ_1.

A discrete *semi-measure* (or simply a *distribution*) on Σ is a function $\mathbb{P} : \Sigma^* \to [0, 1]$ such that $\sum_{\sigma\in\Sigma^*} \mathbb{P}(\sigma) \leq 1$. When this sum is equal to 1 we call $\mathbb{P}$ a *probability measure*. We also consider associated measure functions defined on positive integers $\mu : \mathbb{Z}^+ \to [0, 1]$, such that $\sum_{n\in\mathbb{Z}^+} \mu(n) \leq 1$, using the same nomenclature. When dealing with a one-element alphabet we will often think of a^k as a unary notation for k, so that each $\mathbb{P}$ will correspond to an obvious function μ on integers, namely $\mu(k) = \mathbb{P}(a^k)$.

The *support* of a distribution is the set of strings that receive non-zero measure. We say a distribution has *finite support* if its support is finite and omits the empty string.

Finally, given a set $S \subseteq \Sigma^*$, we define the *conditional distribution* $\mathbb{P}(\sigma \mid S)$ as follows:

$$\mathbb{P}(\sigma \mid S) \quad = \quad \frac{\mathbb{P}(\sigma)}{\sum_{\sigma'\in S} \mathbb{P}(\sigma')} \tag{1}$$

when $\sigma \in S$, and $\mathbb{P}(\sigma \mid S) = 0$ otherwise. As a special case, if $S^+ = \{\sigma \in \Sigma^* : \mathbb{P}(\sigma) > 0\}$ is the support of $\mathbb{P}$, then $\mathbb{P}(\sigma \mid S^+)$ is a probability measure, the *normalization* of $\mathbb{P}$. Given the definition in (1), conditioning on the trivial proposition Σ^* also produces the normalization: $\mathbb{P}(\sigma \mid \Sigma^*) = \mathbb{P}(\sigma \mid S^+)$. More generally, for any S we have $\mathbb{P}(\sigma \mid S) = \mathbb{P}(\sigma \mid S \cap S^+)$. Any conditional distribution $\mathbb{P}(\cdot \mid S)$ will be a probability measure, even if $\mathbb{P}$ is not.

2.2 Probability Generating Functions

From a semi-measure μ we derive a *probability generating function* (pgf) $\mathfrak{G}_\mu$ defined so that $\mathfrak{G}_\mu(z) = \sum_{k=0}^{\infty} \mu(k) z^k$. $\mathfrak{G}_\mu$ essentially summarizes the distribution as a formal power series. We say $\mathfrak{G}_\mu$ is *algebraic* if $y = \mathfrak{G}_\mu(z)$ is a solution to a polynomial equation $0 = Q(y, z)$. We call $\mathfrak{G}_\mu$ *rational* when Q is of degree 1 in y, i.e., when there are polynomials $Q_0(z)$ and $Q_1(z)$ such that $\mathfrak{G}_\mu(z) = \frac{Q_0(z)}{Q_1(z)}$. For example, the pgf for a simple geometric function $\mu(k) = 2^{-k}$ is rational, equal to $\frac{1}{2-z}$. Finally, a pgf is *transcendental* if it is not algebraic. For instance, the pgf for a Poisson distribution (Example 1) is $e^{\lambda z - \lambda}$, easily shown to be transcendental.

For another notable example, consider a semi-measure μ such that $\mu(2^k) = 2^{-k}$ for $k > 0$, and $\mu(n) = 0$ for all other n. Let us write the pgf for μ as $\mathfrak{G}_\mu(z) = \sum_{k=0}^{\infty} c_k z^k$ where $c_k = k^{-1}$ if k is a power of 2, and $c_k = 0$ otherwise. The *Hadamard product* of two power series $\sum_{k=0}^{\infty} c_k z^k$ and $\sum_{k=0}^{\infty} d_k z^k$ is the power series $\sum_{k=0}^{\infty} (c_k d_k) z^k$. One can show that the Hadamard product of a rational and an algebraic power series must be algebraic (Jungen, 1931; Flajolet and Sedgewick, 2001). We thus obtain:

Lemma 1. *The pgf $\mathfrak{G}_\mu(z)$ is transcendental.*

Proof. It is known that the "lacunary" power series $h(z) = \sum_{k=1}^{\infty} z^{2^k}$ is transcendental (e.g., Theorem 1.1.2 of Nishioka 1996). The power series $g(z) = \sum_{k=0}^{\infty} k z^k = z/(z^2 - 2z + 1)$ is patently rational. However, the Hadamard product of $\mathfrak{G}_\mu(z)$ and $g(z)$ gives exactly $h(z)$. Hence $\mathfrak{G}_\mu(z)$ cannot be algebraic. □

When $\Sigma = \{a\}$ we will slightly abuse terminology by speaking of the pgf for a distribution on Σ^* as being rational, algebraic, or transcendental, under the unary encoding.

2.3 Grammars and Probabilistic Grammars

A grammar is a quadruple $\mathcal{G} = (\mathcal{N}, \Sigma, \Pi, X_0)$, given by a finite set of non-terminal symbols $\mathcal{N}$, including a start symbol X_0, an alphabet Σ, and a finite set Π of productions $(\alpha \to \beta)$ where $\alpha, \beta \in (\mathcal{N} \cup \Sigma)^*$. We will refer to elements of Σ^* as *words*.

The standard hierarchy is defined as in Chomsky (1959) (indexed grammars will be introduced separately in §6):

- **Regular (Type 3) Grammars**: All productions are of the form $(X \to \sigma Y)$ or $(X \to \sigma)$, with $\sigma \in \Sigma^*$, $X, Y \in \mathcal{N}$.
- **Context-Free (Type 2) Grammars**: All production rules are of the form $(X \to \alpha)$ with $X \in \mathcal{N}$ and $\alpha \in (\mathcal{N} \cup \Sigma)^*$.
- **Context-Sensitive (Type 1) Grammars**: Productions are of the form $(\alpha X \beta \to \alpha \gamma \beta)$, where $X \in \mathcal{N}$, $\alpha, \beta, \gamma \in (\mathcal{N} \cup \Sigma)^*$, and $\gamma \neq \epsilon$; we also allow $(X_0 \to \epsilon)$ provided X_0 does not occur on the right-hand-side of any production.

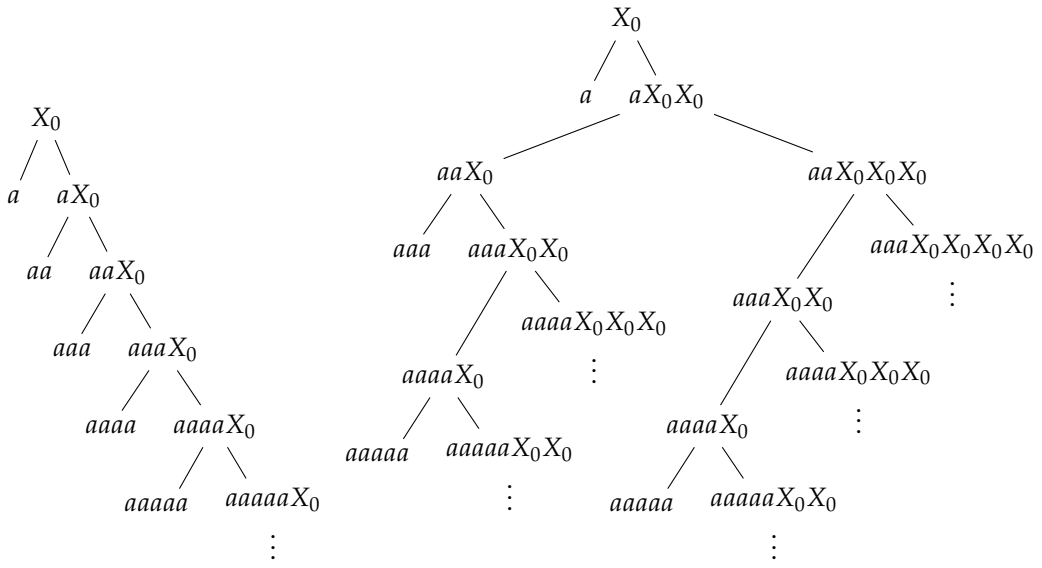

Figure 2: Partial parse trees for Examples 5 (left) and 6 (right).

- **Unrestricted (Type 0) Grammars**: No restrictions.

A *probabilistic grammar* is one with the following property: for each $\alpha \in (\mathcal{N} \cup \Sigma)^*$ there may be only 0, 1, or 2 distinct β with $(\alpha \to \beta) \in \Pi$. If there is 1 we write $\alpha \xrightarrow{1} \beta$, with the interpretation that α rewrites to β with probability 1; if there are 2 we will write $\alpha \xrightarrow{1/2} \beta$ for each, with the interpretation that α rewrites to β with probability $1/2$. Intuitively, when rewriting α we imagine randomly choosing from among those productions with α on the left-hand-side. In effect this means we are only considering probabilistic grammars with production probabilities of $1/2$. As far as definability is concerned, however, this implies no loss relative to allowing arbitrary rational probabilities (Theorem 8 below).

It will often be convenient to refer to production probabilities, which we write as $P_{\mathcal{G}}(\alpha \to \beta)$—or simply $P(\alpha \to \beta)$ when $\mathcal{G}$ is clear from context—always taking on value 0, 1, or $1/2$.

A *parse* of a word σ from a string γ in a grammar $\mathcal{G}$ is a finite sequence $\rho = \langle \rho_0, \ldots, \rho_n \rangle$, where $\rho_0 = \gamma$, $\rho_n = \sigma$, and for all $i < n$: ρ_{i+1} is obtained from ρ_i by replacing the substring α of ρ_i with β, where $(\alpha \to \beta) \in \Pi$, and α is the highest priority substring of ρ_i with an applicable production. The probability of the parse, $P_{\mathcal{G}}(\rho)$, is given by the product of probabilities of productions used in ρ. The probability of γ rewriting as a word σ is given by the sum of probabilities of its parses: $\mathbb{P}^{\gamma}_{\mathcal{G}}(\sigma) = \sum_{\rho:\rho_0=\gamma,\rho_n=\sigma} P_{\mathcal{G}}(\rho)$. We will write $\mathbb{P}_{\mathcal{G}}(\sigma)$ for the distribution $\mathbb{P}^{X_0}_{\mathcal{G}}(\sigma)$ from the start symbol X_0.

Example 5. Consider the following simple probabilistic regular grammar $\mathcal{G}_1$:

$$X_0 \xrightarrow{1/2} aX_0 \qquad X_0 \xrightarrow{1/2} a$$

The tree in Figure 2 depicts all parses of the first five strings, showing that $\mathbb{P}_{\mathcal{G}_1}(a^k) = 2^{-k}$.

Example 6. The following is a probabilistic context-free grammar $\mathcal{G}_2$, only minimally more complex than the regular $\mathcal{G}_1$ by allowing two non-terminals on the right:

$$X_0 \overset{1/2}{\to} aX_0X_0 \qquad X_0 \overset{1/2}{\to} a$$

As revealed in Figure 2, we have $\mathbb{P}_{\mathcal{G}_2}(a) = 2^{-1}$ and $\mathbb{P}_{\mathcal{G}_2}(a^3) = 2^{-3}$, like in Example 5. However, a^5 has two parses, each with probability 2^{-5}. Hence, $\mathbb{P}_{\mathcal{G}_2}(a^5) = 2^{-5} \cdot 2 = 2^{-4}$.

$\mathcal{G}_1$ and $\mathcal{G}_2$ are also both context-sensitive grammars.

Example 7. Finally, we offer an example of a probabilistic unrestricted grammar $\mathcal{G}_3$, which is not regular, context-free, or context-sensitive:

$$\begin{array}{llll} X_0 \overset{1}{\to} WYaZ & YZ \overset{1/2}{\to} U & aV \overset{1}{\to} Va & aU \overset{1}{\to} Ua \\ Ya \overset{1}{\to} aaY & YZ \overset{1/2}{\to} VZ & WV \overset{1}{\to} WY & WU \overset{1}{\to} \epsilon \end{array}$$

With probability 1, X_0 rewrites to $WaaYZ$. Then, since YZ randomly rewrites to either U or VZ, we have that $WaaYZ$ rewrites to either $WaaU$ or $WaaVZ$, each with probability $1/2$. Following the sequence of productions, $WaaU$ rewrites with probability 1 to aa, while $WaaVZ$ rewrites with probability 1 to $WaaaaYZ$. The latter string then again rewrites with probability $1/2$ to $aaaa$ and with probability $1/2$ to $WaaaaaaaaYZ$. In other words, this grammar defines the distribution $\mathbb{P}_{\mathcal{G}_3}(a^{2^k}) = 2^{-k}$, shown above (Lemma 1) to be transcendental.

Example 7 also provides a simple instance of a conditional distribution, since we have $\mathbb{P}_{\mathcal{G}_3}(\sigma) = \mathbb{P}_{\mathcal{G}_1}(\sigma \mid \{a^{2^k} : k > 0\})$.

3 Probabilistic Unrestricted Grammars

Similar to the classical case, probabilistic grammars in general correspond to *probabilistic Turing machines* (de Leeuw et al., 1956), conceived as stochastic word generators. Such models have arguably delineated the class of possible psychological models for as long as the mind has been likened to a computing device (Turing, 1950; Putnam, 1967).

A probabilistic Turing machine (PTM) comes with an infinite one-way read-only random bit tape with Bernoulli($1/2$)-distributed binary variables, as well as a one-way infinite read/write tape initially consisting of a special end symbol $\triangleright$ followed by an infinite list of blank symbols $\sqcup$. Given an alphabet Σ, a PTM consists of finitely many states $S = \{s_0, s_1 \ldots, s_n, s_\#\}$, with a distinguished start state s_0 and a distinguished end state $s_\#$, and finitely many rules of the form:

$$\langle s_i, a_1, b, a_2, d, s_j \rangle$$

where $s_i, s_j \in S$, $a_1, a_2 \in \Sigma \cup \{\triangleright, \sqcup\}$, $b \in \{0, 1\}$, $d \in \{L, R\}$. Such a rule is read: if in state s_i, reading a_1 on the read/write tape and b on the random bit tape, rewrite a_1 as a_2 and go left (if $d = L$) or right (if $d = R$), entering state s_j. We assume for each triple s_i, a_1, b, there is at most one triple a_2, d, s_j such that $\langle s_i, a_1, b, a_2, d, s_j \rangle$ is a rule of $\mathcal{T}$. That is, $\mathcal{T}$ is deterministic given its random input. We also assume that $a_1 = \triangleright$ if and only if $a_2 = \triangleright$; moreover, in this case $d = R$. The machine begins reading $\triangleright$ and halts upon reaching state $s_\#$. By a familiar argument, we can assume that every PTM is in a normal form guaranteeing that,

upon halting, following the $\triangleright$ symbol is some $\sigma \in \Sigma^*$, followed again by an infinite list of blank symbols. The word σ is the official output.

We use $\mathbb{P}_{\mathcal{T}}$ to refer to the distribution effected by $\mathcal{T}$. That is, $\mathbb{P}_{\mathcal{T}}(\sigma)$ is the probability that $\mathcal{T}$ halts with σ as output on the tape. Clearly $\mathbb{P}_{\mathcal{T}}$ is a semi-measure. The proof of the next proposition is similar to the non-probabilistic case (e.g., Chomsky 1959; Minsky 1967); we include the details as later results (Propositions 27 and 29) will depend on them.

Proposition 2. *Probabilistic unrestricted grammars and probabilistic Turing machines define the same distributions.*

Proof. We need to show that $\mathbb{P}_{\mathcal{T}}$ for any PTM $\mathcal{T}$ can be defined by some grammar $\mathcal{G}$. If $\mathcal{T}$ has states $s_0, s_1, \ldots, s_n, s_{n+1}(=s_\#)$, then $\mathcal{G}$ will have non-terminals $X_0, X_1, \ldots, X_n, X_{n+1}$, in addition to $\triangleright$ and $\sqcup$, and one more "loop" non-terminal Λ having 0 productions. Our aim is to mimic the behavior of $\mathcal{T}$, step-by-step, using the non-terminals X_i to keep track of the position of the read/write head. We begin with a rule $(X_0 \rightarrow X_0\triangleright)$ rewriting X as $X_0\triangleright$. Then, for each rule of the form $\langle s_i, a_1, b, a_2, L, s_j\rangle$ of $\mathcal{T}$ we add to $\mathcal{G}$ production rules for all $a \in \Sigma \cup \{\triangleright, \sqcup\}$:

$$aX_ia_1 \quad \rightarrow \quad X_jaa_2$$

If there is no rule for $s_i, a_1, 1-b$ then we also add:

$$aX_ia_1 \quad \rightarrow \quad \Lambda$$

For each rule $\langle s_i, a_1, b, a_2, R, s_j\rangle$, we include a production:

$$X_ia_1 \quad \rightarrow \quad a_2X_j$$

When $a_1 = \sqcup$ we additionally include:

$$X_i \quad \rightarrow \quad a_2X_j$$

Given our priority order on substrings (§2.1), this production will only be used when X_i occurs at the very end of the string. Again, for both of the previous we include productions leading to Λ if there is no rule appropriate for $s_i, a_1, 1-b$.

Finally, we include "clean-up" productions for the stage when we reach X_{n+1}, corresponding to $s_\#$. For all $a \in \Sigma$:

$$\begin{aligned} aX_{n+1} &\rightarrow X_{n+1}a \\ \triangleright X_{n+1} &\rightarrow X_{n+1} \\ X_{n+1}a &\rightarrow aX_{n+1} \\ X_{n+1}\sqcup &\rightarrow X_{n+1} \\ X_{n+1} &\rightarrow \epsilon \end{aligned}$$

That is, (again, due to our priority order on substrings) X_{n+1} moves along the string to the left (which, given our PTM normal form, contains no blank symbols) until it hits $\triangleright$. After erasing it, we move along the string to the right until reaching blank symbols, eliminate them, and then X_{n+1} rewrites to the empty string, giving us a word from Σ^*.

To show that a PTM can simulate any probabilistic grammar is routine (again similar to the classical case, e.g., Minsky 1967). □

These distributions can be given a precise formulation. Call a semi-measure $\mathbb{P}$ *(computably) enumerable* if it is approximable from below (Zvonkin and Levin, 1970); that is, if for each $\sigma \in \Sigma^*$ there is a computably enumerable weakly increasing sequence $q_0, q_1, q_2, \ldots$ of rational numbers, such that $\lim_{i\to\infty} q_i = \mathbb{P}(\sigma)$. Enumerable semi-measures have been argued to constitute a useful idealized inductive target for psychological learning models (Vitányi and Chater, 2017). $\mathbb{P}_{\mathcal{T}}$ and (by Theorem 2) $\mathbb{P}_{\mathcal{G}}$ are always guaranteed to be enumerable: consider the set B_i of binary strings β with $|\beta| \leq i$, such that $\mathcal{T}$ accesses (exactly) the bits of β before terminating with output σ. Letting $q_i = \sum_{\beta\in B_i} 2^{-|\beta|}$, it is then evident that $\lim_{i\to\infty} q_i = \mathbb{P}_{\mathcal{T}}(\sigma)$. In fact, we have the converse as well.

Theorem 3 (Icard 2017a)**.** *Probabilistic grammars (equivalently, probabilistic Turing machines) define exactly the enumerable distributions.*

Proof. Let $\mathbb{P}$ be an enumerable semi-measure on Σ^*. That is, for each word $\sigma \in \Sigma^*$, there is a computably enumerable weakly increasing sequence $q_0, q_1, q_2, \ldots$ of rational numbers such that $\lim_{i\to\infty} q_i = \mathbb{P}(\sigma)$. Assume without loss that $q_0 = 0$. Note then that $\mathbb{P}(\sigma) = \sum_{i=0}^{\infty}(q_{i+1} - q_i)$. Our aim is to show that $\mathbb{P} = \mathbb{P}_{\mathcal{T}}$ for some PTM $\mathcal{T}$.

Let $\langle \_, \_ \rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be a fixed (computable) bijective pairing function with first projection $\pi_1(n) = k$ when $n = \langle k, i\rangle$. Let $\sigma_0, \sigma_1, \sigma_2, \ldots$ be a fixed enumeration of Σ^*, each with a fixed enumeration of approximating rationals $q_0^k, q_1^k, \ldots$ converging from below to $\mathbb{P}(\sigma_k)$. We define a sequence of rational (thus computable) numbers as follows:

$$\begin{array}{rcl} r_0 & = & q_0^0 \\ r_{n+1} & = & r_n + (q_{i+1}^k - q_i^k) \end{array}$$

where we assume $0 = \langle 0, 0\rangle$ and $n + 1 = \langle k, i\rangle$.

Our PTM $\mathcal{T}$ works in stages, observing a random sequence of bits $b_0, \ldots, b_{j-1}$—which we can think of as ever closer approximations to some random real number—while producing an enumeration $r_0, \ldots, r_{j-1}$. At each stage j, we observe a bit b_j and add a rational r_j, then check whether, for any n with $0 \leq n < j$, the following condition (2) is satisfied:

$$r_n < \sum_{i=0}^{j} b_i 2^{-i} - 2^{-j} \text{ and } r_{n+1} > \sum_{i=0}^{j} b_i 2^{-i} + 2^{-j} \tag{2}$$

That is, where $\tilde{p} = \sum_{i=0}^{j} b_i 2^{-i}$ is the rational generated so far, we know our randomly generated real number will lie somewhere in the interval $(\tilde{p} - \epsilon, \tilde{p} + \epsilon)$, and (2) tells us that this interval sits inside the interval (r_n, r_{n+1}). If this holds, output $\sigma_{\pi_1(n+1)}$. Otherwise, move on to stage $j + 1$.

Each word σ has its mass $\mathbb{P}(\sigma)$ distributed across different intervals in $[0, 1]$:

$$\begin{array}{rcl} \mathbb{P}(\sigma_k) & = & \displaystyle\sum_{n:\pi_1(n+1)=k} r_{n+1} - r_n \\ & = & \displaystyle\sum_{i=0}^{\infty}(q_{i+1}^k - q_i^k). \end{array}$$

The procedure generates approximations $\tilde{p} = \sum_{i=0}^{j} b_i 2^{-i}$ to a random real number, and as soon as we are guaranteed that this random number is in one of our intervals between

r_n and $r_{n+1} = r_n + (q^k_{i+1} - q^k_i)$, i.e., that no further bits will take us out of that interval (condition (2) above), we halt and output the string σ_k corresponding to the interval, with $k = \pi_1(n+1)$. The probability of outputting σ is exactly $\mathbb{P}(\sigma)$, and the probability of not halting at all is $1 - \sum_\sigma \mathbb{P}(\sigma)$. □

More restrictive than the class of enumerable semi-measures is the class of *computable* semi-measures (de Leeuw et al., 1956), namely those for which $\mathbb{P}(\sigma)$ can also be computably approximated from above. To be computable we additionally require a weakly *decreasing* sequence of rationals converging to $\mathbb{P}$.

Lemma 4. *Every semi-computable probability measure is also computable.*

Proof. Since $\sum_{\sigma \in \Sigma^*} \mathbb{P}(\sigma) = 1$, we can approximate from below the sum $\sum_{\sigma' \neq \sigma} \mathbb{P}(\sigma')$ by dovetailing through approximating sequences for all strings other than σ. This gives us a sequence $q_1 \leq q_2 \leq q_3, \ldots$ converging from below to $\sum_{\sigma' \neq \sigma} \mathbb{P}(\sigma')$. Thus, the sequence $1 - q_1 \geq 1 - q_2 \geq 1 - q_3, \ldots$ converges from above to $\mathbb{P}(\sigma)$. □

Theorem 3, together with Lemma 4, gives us the following known corollary (see, e.g., Dal Lago and Zorzi 2012; Freer et al. 2014):

Corollary 5. *Almost-surely terminating grammars define the computable probability measures.*

As in the classical case, determining almost-sure termination is algorithmically undecidable. In fact, in the present setting it is even more difficult than the classical halting problem (Kaminski and Katoen, 2015). So there can obviously be no computably defined subclass of grammars or machines corresponding exactly to the computable distributions. To give a sense of how subtle the boundary between almost-sure termination and possible non-termination can be, consider the following example:

Example 8 (Icard 2017a)**.** Imagine a race between a tortoise and a hare. Where `flip(1/4)` is a procedure that returns 1 with probability 1/4 and `Unif(1,7)` returns an integer between 1 and 7 uniformly, consider the following simulation pseudocode:

```
t := 1; h := 0
while (h < t)
    t := t + 1
    if flip(1/4) then h := h + Unif(1,7)
return t-1
```

Whereas this program would almost-surely halt, a small change to the program (e.g., incrementing the tortoise's pace by ϵ) would lead to positive probability of not halting.

Theorem 3 tells us, perhaps unsurprisingly, that probabilistic grammars are capable of defining essentially all of the discrete measures that have been considered in the literature (e.g., all of those surveyed in Johnson et al. 2005), and even encompasses procedures producing values with uncomputable (but enumerable) probabilities, such as Chaitin's famous "halting probability" (Chaitin, 1975). Of course, converting an analytical description into an appropriate procedure can be far from trivial, as illustrated by the following example:

Example 9 (Flajolet et al. 2011)**.** The following pseudocode, compilable into probabilistic Turing machine code, outputs 1 with probability exactly $1/\pi$:

```
x_1, x_2 := Geom(1/4)
t := x_1 + x_2
if flip(5/9) then t := t + 1
for j = 1,2,3
    draw 2t fair coin flips
    if #Heads ≠ #Tails then return 0
return 1
```

The verification depends on an identity for $1/\pi$ due to Ramanujan.

Theorem 3 and Corollary 5 delineate our subject matter, giving an upper bound on the types of discrete distributions probabilistic grammars and machines can possibly represent.

3.1 Conditioning probabilistic grammars

One of the most important operations on probability distributions is conditioning, typically used to encode the effect of updating the distribution with some observation. Our first result about conditioning is negative (cf. Wood et al. 2011 for a related result in the context of so called Solomonoff induction):

Theorem 6. *The class of distributions defined by probabilistic grammars (equivalently, probabilistic Turing machines) is not closed under conditioning, even with finite sets.*

Proof. Take any computably enumerable (but not computable) real number p and rational number r such that $p + r < 1$. By Theorem 3 there is a grammar $\mathcal{G}$ that terminates with probability $p + r$, such that $\mathbb{P}_{\mathcal{G}}(a) = p$ and $\mathbb{P}_{\mathcal{G}}(aa) = r$. Consider the support $S^+ = \{a, aa\}$ of $\mathbb{P}_{\mathcal{G}}$. We claim that the normalization $\mathbb{P}_{\mathcal{G}}(\sigma \mid S^+)$ is not computably enumerable. If it were, it would also be computable (Lemma 4). In particular, there would be a computable sequence $q_1 \geq q_2 \geq q_3 \ldots$ converging from above to $\mathbb{P}_{\mathcal{G}}(a \mid S^+) = \frac{p}{p+r}$. But from this we can obtain another sequence $\frac{t_1}{t_1+r} \geq \frac{t_2}{t_2+r} \geq \frac{t_3}{t_3+r} \ldots$ converging to $\frac{p}{p+r}$, such that the sequence $t_1, t_2, t_3 \ldots$ converges from above to p, which by hypothesis is impossible. □

At the same time, we obtain a positive closure result by restricting attention to those grammars or machines that almost-surely terminate, or slightly more generally to those that define computable distributions. In fact, as Freer et al. (2014) describe, there is a single probabilistic Turing machine that takes as input (the code of) a machine $\mathcal{T}$ and (the code of) a computable predicate S, and then produces strings σ with probabilities $\mathbb{P}_{\mathcal{T}}(\sigma \mid S)$. In the present setting, all this machine must do is implement a kind of "rejection sampling" procedure, repeatedly running $\mathcal{T}$ until it outputs some $\sigma \in S$, at that point returning σ.

Theorem 7. *The computable semi-measures are closed under conditioning with computable sets.*

4 Probabilistic Regular Grammars

We turn now to the most restrictive class of grammars, the probabilistic regular grammars (PRGs). These grammars have received relatively little direct attention. However, due

to their equivalence with probabilistic finite state automata (see, e.g., Smith and Johnson 2007), which are in turn equivalent in expressive power to discrete hidden Markov models (Dupont et al., 2005) as well as so called discrete phase-type distributions (O'Cinneide, 1990), much is already known about them. Such models furthermore have a long history in psychology and cognitive science (Miller, 1952).

As above, we deal with the special case of only $1/2$ production probabilities, which would correspond to $1/2$ transition probabilities in probabilistic automata.

4.1 Expressing Rational Probabilities

A first observation is that, had we begun with any real-valued probabilities q and $1-q$, we would nonetheless be able to simulate $1/2$ productions. In this sense the assumption that we do have $1/2$ productions is with no loss. The trick goes back to von Neumann (1951) and can in fact be carried out with PRGs. For example, if we wanted $X \overset{1/2}{\to} Y_1$ and $X \overset{1/2}{\to} Y_2$, we could simply add two new non-terminals, Z_1, Z_2, and define:

$$\begin{array}{lll} X \overset{q}{\to} Z_1 & Z_1 \overset{q}{\to} X & Z_2 \overset{1-q}{\to} X \\ X \overset{1-q}{\to} Z_2 & Z_1 \overset{1-q}{\to} Y_1 & Z_2 \overset{q}{\to} Y_2 \end{array}$$

Perhaps more impressively, PRGs are able to produce strings with arbitrary rational probabilities using only $1/2$. (This basic idea can be traced back at least to Knuth and Yao 1976.)

Example 10. To simulate $X \overset{1/3}{\to} Y_1$, $X \overset{1/3}{\to} Y_2$ and $X \overset{1/3}{\to} Y_3$ we again add just two new non-terminals and define:

$$\begin{array}{lll} X \overset{1/2}{\to} Z_1 & Z_1 \overset{1/2}{\to} X & Z_2 \overset{1/2}{\to} Y_2 \\ X \overset{1/2}{\to} Z_2 & Z_1 \overset{1/2}{\to} Y_1 & Z_2 \overset{1/2}{\to} Y_3 \end{array}$$

The probability of X rewriting to each is $\sum_{n>0} 1/2^{2n} = 1/3$.

Indeed, consider any rational number q, guaranteed to have a periodic binary expansion $0.b_1 \dots b_k \overline{b_{k+1} \dots b_{k+m}}$. To simulate productions $(X \overset{q}{\to} Y_1), (X \overset{1-q}{\to} Y_0)$, we use non-terminals $X_1 (= X), X_2, \dots, X_{k+m}$ and introduce productions:

$$\begin{array}{ll} X_i \overset{1/2}{\to} Y_{b_i} & \text{for each } i \leq k+m \\ X_i \overset{1/2}{\to} X_{i+1} & \text{for each } i < k+m \\ X_{k+m} \overset{1/2}{\to} X_{k+1} & \end{array}$$

X rewrites to Y_1 with probability q and to Y_0 with probability $1-q$. The intuition is similar to the proof of Theorem 3: we are effectively generating ever closer approximations to a random real number, and once we definitively undershoot q (by generating an nth bit 0 when the nth bit of q is 1) or overshoot q (generating 1 when the corresponding bit of q is 0) we know which of Y_1 and Y_0 to choose.

Repeating this procedure finitely many times gives us:

Theorem 8. *Let $\mathbb{P}_1, \dots, \mathbb{P}_n$ be PRG-definable distributions. For any rational numbers $q_1, \dots, q_n$, with $\sum_{i \leq n} q_i \leq 1$, the semi-measure $\mathbb{P}$ given by $\mathbb{P}(\sigma) = \sum_{i \leq n} q_i \mathbb{P}_i(\sigma)$ is also PRG-definable.*

From now on we will freely label rules with rational probabilities, with the understanding that they can be rewritten using only $1/2$, perhaps by adding further non-terminals.

Corollary 9. *Probabilistic regular grammars define all finite-support rational-valued distributions.*

This already captures an extraordinarily wide variety of distributions, including many that are of central importance in psychological modeling such as (rational parameter valued) Bayesian networks (Pearl, 1988). We also have:

Example 11 (Beta-Binomial)**.** Recall Example 4. As long as $\alpha, \beta \in \mathbb{Z}^+$, the (finitely many non-zero) probability values will always be rational, which by Corollary 9 means we can define it with a PRG. The more general Dirichlet-Multinomial can likewise be expressed, again provided we restrict to positive-integer-valued parameters.

In addition, PRGs can define many important infinite-support distributions, following again from Theorem 8. Recall Example 2:

Example 12 (Negative Binomial)**.** Let us introduce $t+1$ non-terminals $\mathcal{N} = \{X_0, \ldots, X_t\}$ with productions for each $n < t$:

$$X_n \overset{1-q}{\to} aX_n \qquad X_n \overset{q}{\to} X_{n+1} \qquad X_t \overset{1-q}{\to} aX_t \qquad X_t \overset{q}{\to} \epsilon$$

This easily defines the negative binomial distribution $\mu_{q,t}$ from Example 2.

Any discrete probability can of course be arbitrarily well approximated by rational-valued distributions, and the same is even true for continuous measures: the finite-support rational-valued distributions are everywhere dense in the space of Borel probability measures (under the weak topology, see Billingsley 1999). Moreover, this extends to the setting of computable probability spaces, where we additionally require approximating sequences to be uniformly computable (Gács, 2005; Ackerman et al., 2019). Thus, concerning pure expressivity, if we accept mere approximation (which we must in the continuous setting anyway), probabilistic regular grammars with $1/2$-productions suffice in principle.

4.2 Normal Form and Matrix Representation

We would nonetheless like to understand the expressive limitations of PRGs, and in this direction it will be convenient to assume a normal form for them.

In the following lemma let us say that a non-terminal Y is *transient* if either there are are no productions for Y at all, or there is $\alpha \in \Sigma^*$ such that $(Y \to \alpha) \in \Pi$. In other words, Y fails to be transient just when it rewrites with probability 1 to another non-terminal. Say that Y is *reachable* from X if either $X = Y$ or there is some sequence $(X \to Z_1), \ldots, (Z_n \to Y)$.

Lemma 10 (PRG normal form)**.** *Every PRG-definable distribution can be expressed by a PRG with the following properties:*

1. *In all productions $(X \to \sigma Y)$ and $(X \to \sigma)$ we have $|\sigma| \leq 1$, i.e., either $\sigma \in \Sigma$ or $\sigma = \epsilon$.*
2. *X_0 does not appear on the right-hand-side of any production, and there are only productions of the form $(X_0 \to Y)$ with Y a non-terminal.*
3. *From every X some transient Y is reachable.*

Proof. To show 1, it is enough to observe that whenever we have a production $(X \to abY)$ where $a, b \in \Sigma$, we can simply introduce a new non-terminal Z and replace it with two new productions $(X \to aZ)$ and $(Z \to bY)$. By induction we can always replace any such $(X \to \sigma Y)$ with $|\sigma|$ new productions.

To guarantee 2, introduce a new non-terminal Z, replace X_0 with Z everywhere in the grammar, and add $(X_0 \to Z)$.

For 3, consider the set $\mathcal{X}$ of non-terminals reachable from X. If $\mathcal{X}$ contains no transient elements, then for every $Z \in \mathcal{X}$ there are $Z_1, Z_2 \in \mathcal{X}$ with $(Z \to \sigma_1 Z_1), (Z \to \sigma_2 Z_2) \in \Pi$ (possibly $Z_1 = Z_2$). Since the elements of $\mathcal{X}$ thus always rewrite to one another, the distribution on words would not change if we simply removed all productions with elements of $\mathcal{X}$ on the left-hand-side. This immediately guarantees all elements of $\mathcal{X}$ are transient. If $\mathcal{X} = \varnothing$ then X itself is transient. □

From here on let us assume that $\mathcal{G}$ is in normal form, satisfying 1-3 of Lemma 10. We can represent $\mathcal{G}$ using (substochastic) matrices. Where $\mathcal{N} = \{X_0, \ldots, X_m\}$ let $\mathbf{L}$ be an $m \times m$ matrix (indexed $1, \ldots, m$), with entries determined by $\mathcal{G}$:

$$\mathbf{L}[i,j] \quad = \quad P(X_i \to X_j)$$

Similarly for each $a \in \Sigma$ we define an $m \times m$ matrix $\mathbf{A}$:

$$\mathbf{A}[i,j] \quad = \quad P(X_i \to aX_j)$$

To summarize the remaining productions we define a (row) vector $\mathbf{v}$ and (column) vectors $\mathbf{w}$ and $\mathbf{a}$ (one for each $a \in \Sigma$), all of length m, defined so that:

$$\begin{aligned} \mathbf{v}[i] &= P(X_0 \to X_i) \\ \mathbf{w}[i] &= P(X_i \to \epsilon) \\ \mathbf{a}[i] &= P(X_i \to a) \end{aligned}$$

The probability of a word $\sigma = a_1 \ldots a_n$ can thus be represented:

$$\begin{aligned} \mathbb{P}_{\mathcal{G}}(\sigma) &= \sum_{k_1,\ldots,k_{n+1} \geq 0} \mathbf{v}\mathbf{L}^{k_1}\mathbf{A}_1 \ldots \mathbf{L}^{k_n}\mathbf{A}_n\mathbf{L}^{k_{n+1}}\mathbf{w} + \sum_{k_1,\ldots,k_n \geq 0} \mathbf{v}\mathbf{L}^{k_1}\mathbf{A}_1 \ldots \mathbf{L}^{k_n}\mathbf{a}_n \\ &= \mathbf{v}\big(\sum_{k \geq 0} \mathbf{L}^k\big)\mathbf{A}_1 \ldots \big(\sum_{k \geq 0} \mathbf{L}^k\big)\mathbf{A}_n\big(\sum_{k \geq 0} \mathbf{L}^k\big)\mathbf{w} + \mathbf{v}\big(\sum_{k \geq 0} \mathbf{L}^k\big)\mathbf{A}_1 \ldots \big(\sum_{k \geq 0} \mathbf{L}^k\big)\mathbf{a}_n \end{aligned}$$

The fact that $\mathcal{G}$ is in normal form guarantees that these infinite sums can be eliminated from the equation. Specifically:

Lemma 11. $\lim_{k \to \infty} \mathbf{L}^k = \mathbf{0}$.

Proof. The statement is equivalent to the spectral radius of $\mathbf{L}$ (largest absolute value of its eigenvalues) being strictly less than 1 (see, e.g., Kress 1998). Because $\mathbf{L}$ is a contraction (e.g., with respect to l_1 norm) we know the spectral radius is less than or equal to 1, so it remains only to show that it cannot be 1.

If 1 were an eigenvalue of $\mathbf{L}$ then we would have $\mathbf{L}\mathbf{x} = \mathbf{x}$ for some non-zero $\mathbf{x}$. Thus, for each index i of $\mathbf{x}$ one of the following must hold: (1) $\mathbf{x}[i] = 0$, (2) $\mathbf{x}[i] = \frac{1}{2}\mathbf{x}[j]$ for some j, (3) $\mathbf{x}[i] = \frac{1}{2}\mathbf{x}[j] + \frac{1}{2}\mathbf{x}[k]$ for some j, k, or (4) $\mathbf{x}[i] = \mathbf{x}[j]$ for some j. Note that those i corresponding to transient non-terminals X_i will never satisfy (3) or (4). Consider the set

$I = \{i$: the absolute value of $\mathbf{x}[i]$ is maximal$\}$. Since $\mathbf{x}$ is non-zero, no $i \in I$ can satisfy (1) or (2). Moreover, for no such i do we have $\mathbf{x}[i] = \mathbf{x}[j]$ with $j \notin I$. By condition 3 of Lemma 10 (normal form), for some i we must therefore have $\mathbf{x}[i] = \frac{1}{2}\mathbf{x}[j] + \frac{1}{2}\mathbf{x}[k]$ for some j, k, at least one of which must not be in I. But this too is impossible. □

Lemma 12. $\sum_{k\geq 0} \mathbf{L}^k = (\mathbf{I} - \mathbf{L})^{-1}$

Proof. The proof is standard, but we give it for completeness. We want to show that the matrix product $(\mathbf{I} - \mathbf{L})(\sum_{k\geq 0} \mathbf{L}^k)$ is equal again to the identity matrix $\mathbf{I}$.

$$\begin{aligned}
(\mathbf{I} - \mathbf{L})(\sum_{k\geq 0} \mathbf{L}^k) &= \lim_{n\to\infty} \Big((\mathbf{I} - \mathbf{L})(\sum_{n\geq k\geq 0} \mathbf{L}^k) \Big) \\
&= \lim_{n\to\infty} \Big(\sum_{n\geq k\geq 0} \mathbf{L}^k - \sum_{n\geq k\geq 0} \mathbf{L}^{k+1} \Big) \\
&= \lim_{n\to\infty} \big(\mathbf{L}^0 - \mathbf{L}^{n+1} \big) \\
&= \lim_{n\to\infty} \big(\mathbf{I} - \mathbf{L}^{n+1} \big)
\end{aligned}$$

But because $\lim_{n\to\infty} \mathbf{L}^{n+1} = \mathbf{0}$, this limit is in fact equal to $\mathbf{I}$. □

By Lemma 12, and abbreviating $(\mathbf{I} - \mathbf{L})^{-1}$ as $\mathbf{M}$, we have:

$$\begin{aligned}
\mathbb{P}_{\mathcal{G}}(\sigma) &= \mathbf{v}\mathbf{M}\mathbf{A}_1 \dots \mathbf{M}\mathbf{A}_n\mathbf{M}\mathbf{w} + \mathbf{v}\mathbf{M}\mathbf{A}_1 \dots \mathbf{M}\mathbf{a}_n \\
&= \mathbf{v}\mathbf{M}\mathbf{A}_1 \dots \mathbf{M}(\mathbf{A}_n\mathbf{M}\mathbf{w} + \mathbf{a}_n) \qquad (3)
\end{aligned}$$

Because multiplying and taking inverses of matrices never leads to irrational numbers, Eq. (3) establishes a sort of converse of Corollary 9:

Proposition 13. *For any PRG $\mathcal{G}$ and any word σ, the probability $\mathbb{P}_{\mathcal{G}}(\sigma)$ is a rational number.*

4.3 Rational Generating Functions

Theorem 8 and Proposition 13 tell us about the kinds of probability values specific words can take. But we would also like to understand the overall structure of the distribution produced by a PRG. In the present section we restrict attention to $\Sigma = \{a\}$, which we interpret as providing unary notations for positive integers. In this case Eq. 3 can be written even more simply. Abbreviating $\mathbf{MA}$ by $\mathbf{N}$ we have:

$$\mathbb{P}_{\mathcal{G}}(a^{k+1}) = \mathbf{v}\mathbf{N}^k\big(\mathbf{N}\mathbf{M}\mathbf{w} + \mathbf{M}\mathbf{a}\big)$$

In other words, there are fixed vectors $\mathbf{v}$ and $\mathbf{u}$ such that:

$$\mathbb{P}_{\mathcal{G}}(a^{k+1}) = \mathbf{v}\mathbf{N}^k\mathbf{u} \qquad (4)$$

Eq. 4 leads to the following result (cf. Paz 1971):

Lemma 14. *There are fixed constants $c_1, \dots, c_m$, such that for every $k \in \mathbb{N}$:*

$$\mathbb{P}_{\mathcal{G}}(a^{k+m+1}) = \sum_{i=1}^{m} c_i\, \mathbb{P}_{\mathcal{G}}(a^{k+i}).$$

Proof. By the Cayley-Hamilton Theorem, $\mathbf{N}$ satisfies its own characteristic equation, i.e.:

$$\mathbf{N}^m \quad = \quad c_1\mathbf{I} + c_2\mathbf{N} + \cdots + c_m\mathbf{N}^{m-1}$$

for constants $c_1, \ldots, c_m$. Multiplying each term on the left by $\mathbf{v}\mathbf{N}^k$ and on the right by $\mathbf{u}$:

$$\mathbf{v}\mathbf{N}^{k+m}\mathbf{u} \;=\; c_1\mathbf{v}\mathbf{N}^k\mathbf{u} + c_2\mathbf{v}\mathbf{N}^{k+1}\mathbf{u} + \cdots + c_m\mathbf{v}\mathbf{N}^{k+m-1}\mathbf{u}.$$

In other words, $\mathbb{P}_{\mathcal{G}}(a^{k+m+1}) = c_1\,\mathbb{P}_{\mathcal{G}}(a^{k+1}) + c_2\,\mathbb{P}_{\mathcal{G}}(a^{k+2}) + \cdots + c_m\,\mathbb{P}_{\mathcal{G}}(a^{k+m})$. □

A quick, high-level derivation of this next result from Lemma 14 can be found, e.g., in Theorem 4.1.1 of Stanley (2011). We give an explicit proof here.

Theorem 15. *The probability generating function for any PRG on $\Sigma = \{a\}$ is a rational function.*

Proof. The aim is to show there are polynomials $Q_0(z)$ and $Q_1(z)$ such that $\mathfrak{G}_{\mathcal{G}}(z) = \frac{Q_0(z)}{Q_1(z)}$. In other words we want $Q_0(z) - Q_1(z)\mathfrak{G}_{\mathcal{G}}(z) = 0$. We leave off the subscript $\mathcal{G}$ from $\mathfrak{G}_{\mathcal{G}}$ and from $\mathbb{P}_{\mathcal{G}}$. Define $Q_1(z)$ to be the m-degree polynomial $-c_1z^m + -c_2z^{m-1} + \cdots + -c_mz + 1$:

$$Q_0(z) - Q_1(z)\mathfrak{G}(z) \quad = \quad Q_0(z) + c_1z^m\mathfrak{G}(z) + \cdots + c_mz\mathfrak{G}(z) - \mathfrak{G}(z)$$

Setting this equal to 0 we can solve for $Q_0(z)$. We would like $Q_0(z)$ such that:

$$\begin{aligned}
\mathfrak{G}(z) &= Q_0(z) + c_1z^m\mathfrak{G}(z) + \cdots + c_mz\mathfrak{G}(z) \\
\sum_{k=0}^{\infty}\mathbb{P}(a^k)z^k &= Q_0(z) + \sum_{k=0}^{\infty}c_1\mathbb{P}(a^k)z^{k+m} + \cdots + \sum_{k=0}^{\infty}c_m\mathbb{P}(a^k)z^{k+1}
\end{aligned}$$

By Lemma 14 we have $\sum_{k=m}^{\infty}\mathbb{P}(a^k)z^k$ equal to:

$$\sum_{k=0}^{\infty}c_1\mathbb{P}(a^k)z^{k+m} + \cdots + \sum_{k=m-1}^{\infty}c_m\mathbb{P}(a^k)z^{k+1}.$$

Thus, setting $Q_0(z)$ equal to the $m-1$-degree polynomial

$$\mathbb{P}(\epsilon) + \big(\mathbb{P}(a) - c_m\mathbb{P}(\epsilon)\big)z + \cdots + \Big(\mathbb{P}(a^{m-1}) - \big(c_2\mathbb{P}(\epsilon) + c_3\mathbb{P}(a) + \cdots + c_m\mathbb{P}(a^{m-2})\big)\Big)z^{m-1}$$

gives us the desired equality. □

A version of Theorem 15 can be traced back to observations of Schützenberger (1961) (see also Eilenberg 1974; Salomaa and Soittola 1978; Kuich and Salomaa 1986; Flajolet and Sedgewick 2001; Kornai 2008; Bhattiprolu et al. 2017). It is known that probabilistic automata and the closely related discrete phase-type distributions (and therefore also PRGs) do not define all probability distributions with rational pgfs, even if we were to allow arbitrary positive real number weights (Eilenberg, 1974; Soittola, 1976; O'Cinneide, 1990). The reason is that any generating function for one of these devices must be a merge of rational functions possessing unique poles of minimal modulus (i.e., there can only be one zero of the denominator with minimal absolute value). See §VIII Example 6.1 of Eilenberg (1974).

Note that it is important for Theorem 15 that positive integers be represented in unary. If we instead used a binary representation, for example, then as 10^k is the binary representation of 2^k, we could define a distribution whose pgf is transcendental (Lemma 1).

4.4 Conditioning with a Regular Set

Given the previous results we can show that PRG-definable distributions are closed under conditioning with regular sets of words. In that direction we first show the following useful closure result (cf. Nederhof and Satta 2003):

Lemma 16 (Normalization)**.** *Given a PRG $\mathcal{G}$, consider the set $S^+_{\mathcal{G}} = \{\sigma \in \Sigma^* : \mathbb{P}_{\mathcal{G}}(\sigma) > 0\}$ of words with positive probability. There is a PRG $\mathcal{G}^+$ such that $\mathbb{P}_{\mathcal{G}^+}(\sigma) = \mathbb{P}_{\mathcal{G}}(\sigma \mid S^+_{\mathcal{G}})$ for all σ.*

Proof. Given $\mathcal{G}$, for each non-terminal X_i define:

$$\nu(X_i) \quad = \quad \sum_{\sigma \in \Sigma^*} \mathbb{P}^{X_i}_{\mathcal{G}}(\sigma)$$

to be the probability that X_i rewrites to a word. We claim that $\nu(X_i)$ is always a rational number. Consider the grammar $\mathcal{G}$ now with X_i as the start symbol, and assume without loss that this grammar is in normal form (so in particular X_i satisfies condition (2) of Lemma 10). Define row vector $\mathbf{v}$, matrix $\mathbf{T}$, and column vector $\mathbf{f}$ so that:

$$\begin{aligned} \mathbf{v}[j] &= P_{\mathcal{G}}(X_i \to X_j) \\ \mathbf{T}[j,k] &= \sum_{\sigma \in \Sigma \cup \{\epsilon\}} P_{\mathcal{G}}(X_j \to \sigma X_k) \\ \mathbf{f}[j] &= \sum_{\sigma \in \Sigma \cup \{\epsilon\}} P_{\mathcal{G}}(X_j \to \sigma) \end{aligned}$$

Evidently, $\nu(X_i) = \mathbf{v}\big(\sum_{k \geq 0} \mathbf{T}^k\big)\mathbf{f}$, so it remains only to show that $\sum_{k \geq 0} \mathbf{T}^k$ is rational. As $\mathcal{G}$ is in normal form, the same argument as in Lemma 11 establishes that $\lim_{k \to \infty} \mathbf{T}^k = \mathbf{0}$, and hence that $\sum_{k \geq 0} \mathbf{T}^k = (\mathbf{I} - \mathbf{T})^{-1}$ (Lemma 12), implying that $\nu(X_i)$ is indeed rational.

To obtain the "normalized" grammar $\mathcal{G}^+$ we simply multiply each rule probability by a certain rational number, with Theorem 8 guaranteeing that we can always construct such a grammar. Specifically, each rule $(X \to \sigma)$ is now in $\mathcal{G}^+$ given probability

$$\frac{P_{\mathcal{G}}(X \to \sigma)}{\nu(X)}$$

and each rule $(X \to \sigma Y)$ is now given probability

$$\frac{P_{\mathcal{G}}(X \to \sigma Y) \cdot \nu(Y)}{\nu(X)}.$$

To show that $\mathbb{P}_{\mathcal{G}^+}(\sigma) = \mathbb{P}_{\mathcal{G}}(\sigma \mid S^+_{\mathcal{G}})$ we establish a slightly stronger claim, namely that for every non-terminal X and every parse $\rho = \langle X, \ldots, \sigma \rangle$, with $\sigma \in \Sigma^*$, we have

$$P_{\mathcal{G}^+}(\rho) \quad = \quad \frac{P_{\mathcal{G}}(\rho)}{\nu(X)}. \tag{5}$$

We show (5) by induction on the length of ρ, with the base case being $\rho = \langle X, \sigma \rangle$ and $\sigma \in \Sigma \cup \{\epsilon\}$. Then $P_{\mathcal{G}^+}(\langle X, \sigma \rangle) = P_{\mathcal{G}^+}(X \to \sigma) = P_{\mathcal{G}}(X \to \sigma)/\nu(X) = P_{\mathcal{G}}(\langle X, \sigma \rangle)/\nu(X)$.

For the inductive case, consider $\rho = \langle X, \sigma_0 Y, \ldots, \sigma \rangle$, where $\sigma_0 \in \Sigma \cup \{\epsilon\}$ and $\sigma = \sigma_0 \sigma_1$:

$$\begin{aligned} P_{\mathcal{G}^+}(\langle X, \sigma_0 Y, \ldots, \sigma \rangle) &= P_{\mathcal{G}^+}(X \to \sigma_0 Y) \cdot P_{\mathcal{G}^+}(\langle Y, \ldots, \sigma_1 \rangle) \\ &= \frac{P_{\mathcal{G}}(X \to \sigma_0 Y) \cdot \nu(Y)}{\nu(X)} \cdot \frac{P_{\mathcal{G}}(\langle Y, \ldots, \sigma_1 \rangle)}{\nu(Y)} \\ &= \frac{P_{\mathcal{G}}(X \to \sigma_0 Y) \cdot P_{\mathcal{G}}(\langle Y, \ldots, \sigma_1 \rangle)}{\nu(X)} \\ &= \frac{P_{\mathcal{G}}(\rho)}{\nu(X)}. \end{aligned}$$

Taking $X = X_0$ and summing over all parses of σ, this establishes the main claim. □

Recall a *non-deterministic finite-state automaton* (NFA) is a tuple $\mathcal{D} = (\mathcal{Q}, \Sigma, \Delta, q_0, q_f)$ consisting of a set $\mathcal{Q}$ of *states*, an alphabet Σ, a transition relation $\Delta \subseteq \mathcal{Q} \times \Sigma \times \mathcal{Q}$, and distinguished start q_0 and final q_f states (see, e.g., Hopcroft and Ullman 1979). $\mathcal{D}$ *accepts* a word $\sigma \in \Sigma^*$ if there is some sequence of transitions starting in state q_0, successively reading the symbols of σ, and ending in q_f. NFAs accept exactly the *regular sets* of words.

We now show how to construct, from PRG $\mathcal{G}$ and some NFA $\mathcal{D}$ accepting a regular set R, a grammar $\mathcal{G} \otimes \mathcal{D}$ with the property that $\mathbb{P}_{\mathcal{G} \otimes \mathcal{D}}(\sigma \mid S^+_{\mathcal{G} \otimes \mathcal{D}}) = \mathbb{P}(\sigma \mid R)$. Lemma 16 in turn guarantees that this distribution can itself be defined by a PRG, which will prove:

Theorem 17. *The class of distributions definable by probabilistic regular grammars is closed under conditioning with regular sets.*

Proof. The non-terminals of $\mathcal{G} \otimes \mathcal{D}$ are all pairs $\langle X_i, Q \rangle$, with X_i a non-terminal of $\mathcal{G}$ and $Q \subseteq \mathcal{Q}$ a non-empty set of states of $\mathcal{D}$. We also include a "loop" non-terminal Λ. The productions of $\mathcal{G} \otimes \mathcal{D}$ are obtained from $\mathcal{G}$ and $\mathcal{D}$. As before assume $\mathcal{G}$ is in normal form.

1. For each $(X_i \to X_j)$ and Q, add a production $(\langle X_i, Q \rangle \to \langle X_j, Q \rangle)$.
2. For each $(X_i \to \epsilon)$ and Q containing q_f, add a production $(\langle X_i, Q \rangle \to \epsilon)$.
3. For each $(X_i \to a)$ and Q containing a q with $(q, a, q_f) \in \Delta$, add $(\langle X_i, Q \rangle \to a)$.
4. For each $(X_i \to a X_j)$ and Q, if $Q' = \{q' : (q, a, q') \text{ for some } q \in Q\}$ is non-empty, then add the production $(\langle X_i, Q \rangle \to \langle X_j, Q' \rangle)$.

Finally, if in $\mathcal{G}$ there were two productions with X_i on the left hand side, and for some Q we have added fewer than two productions for $\langle X_i, Q \rangle$, add a production $(\langle X_i, Q \rangle \to \Lambda)$. This guarantees, in the resulting grammar $\mathcal{G} \otimes \mathcal{D}$, that every parse of every word has the same probability as it did in $\mathcal{G}$. In other words, $\mathbb{P}_{\mathcal{G}}(\sigma) = \mathbb{P}_{\mathcal{G} \otimes \mathcal{D}}(\sigma)$ for every σ. However, the words that receive positive measure in $\mathcal{G} \otimes \mathcal{D}$ are exactly those in R that received positive measure in $\mathcal{G}$. That is to say, $R \cap S^+_{\mathcal{G}} = S^+_{\mathcal{G} \otimes \mathcal{D}}$. In addition, $\mathbb{P}_{\mathcal{G}}(\sigma \mid R) = \mathbb{P}_{\mathcal{G}}(\sigma \mid R \cap S^+_{\mathcal{G}})$. Hence, $\mathbb{P}_{\mathcal{G} \otimes \mathcal{D}}(\sigma \mid S^+_{\mathcal{G} \otimes \mathcal{D}}) = \mathbb{P}(\sigma \mid R)$, as desired. □

5 Probabilistic Context-Free Grammars

Probabilistic context-free grammars (PCFGs) are undoubtedly the most thoroughly studied of probabilistic grammars. Due to their ability to capture hierarchical structure in language,

and the existence of good learning models, PCFGs have been ubiquitous in computational approaches to language and grammar (see, e.g., Klein and Manning 2003; Levy 2008; Kim et al. 2019). But they have also seen many applications in other areas of psychology, for example, as encoding a probabilistic hypothesis space for concepts (Tenenbaum et al., 2011).

As mentioned above, by results of Abney et al. (1999) and Etessami and Yannakakis (2009), PCFGs express the same class of distributions as pushdown automata and recursive Markov chains. (However, see the discussion below in §5.1 for qualification.)

Example 13 (Symmetric Random Walk)**.** Recall the symmetric 1/2-random walk on the non-negative integers (Example 3). The hitting time for first reaching 0 is a random variable with the same distribution as that defined by the simple grammar $\mathcal{G}_2$ in Example 6:

$$X_0 \overset{1/2}{\to} aX_0X_0 \qquad X_0 \overset{1/2}{\to} a$$

In the analytic expression from Example 3, the kth Catalan number c_k counts the number of parses of a^{2k+1}, while each parse has probability 2^{-2k+1}.

The next four examples reveal irrational word probabilities:

Example 14. Consider the grammar:

$$X_0 \overset{1/4}{\to} X_0X_0 \qquad X_0 \overset{1/4}{\to} a \qquad X_0 \overset{1/2}{\to} \epsilon$$

Then, for instance, $\mathbb{P}(\epsilon)$ is a solution to the equation $x = \frac{1}{4}x^2 + \frac{1}{2}$, equal to $2 - \sqrt{2}$.

Example 15 (Olmedo et al. 2016)**.** The following grammar is quite simple:

$$X_0 \overset{1/2}{\to} X_0X_0X_0 \qquad X_0 \overset{1/2}{\to} \epsilon$$

Yet, $\mathbb{P}(\epsilon)$ is a solution to $x = \frac{1}{2}x^3 + \frac{1}{2}$, equal to $1/\varphi$, where φ is the golden ratio.

Example 16 (Etessami and Yannakakis 2009)**.** This grammar is also quite simple.

$$X_0 \overset{1/6}{\to} X_0X_0X_0X_0X_0 \qquad X_0 \overset{1/2}{\to} a \qquad X_0 \overset{1/3}{\to} \epsilon$$

However, $\mathbb{P}(\epsilon)$ is a solution to $x = \frac{1}{6}x^5 + \frac{1}{3}$, which is is not even solvable by radicals.

The last three examples involve distributions assigning positive probability to the empty string. This is not essential for producing irrational probability values:

Example 17. Consider the grammar:

$$X_0 \overset{1/2}{\to} aY \qquad X_0 \overset{1/2}{\to} a \qquad Y \overset{1/4}{\to} YY \qquad Y \overset{1/4}{\to} a \qquad Y \overset{1/2}{\to} \epsilon$$

Then $\mathbb{P}(\epsilon) = 0$, but, for instance, $\mathbb{P}(a) = \frac{3-\sqrt{2}}{2}$.

However, it might also be noticed that these distributions all maintain infinite support, or assign positive probability to the empty string. It turns out that this is essential:

Proposition 18. *For distributions with finite support, PCFGs generate all and only the rational-valued semi-measures.*

Proof. Suppose a PCFG $\mathcal{G}$ has finite support $S_{\mathcal{G}}^{+} = \{a^{k_1}, \ldots, a^{k_n}\}$ with all $k_i > 0$, and let $m = \max(k_1, \ldots, k_n)$. If X_0 rewrites to some intermediate string $\alpha \in (\mathcal{N} \cup \Sigma)^*$ with positive probability and $|\alpha| > m$, then $|\alpha| - m$ of the non-terminals in α cannot rewrite to a positive-length word with any positive probability. Otherwise the grammar would assign positive measure to a word longer than m. Revise $\mathcal{G}$ by removing each non-terminal Y such that $\mathbb{P}_{\mathcal{G}}^{Y}(\epsilon) = 1$, and replacing Y with the empty string anywhere it appears on the right-hand side of a rule. Thus, in the revised grammar X_0 rewrites with positive probability to only finitely many intermediate strings α that may eventually rewrite to a word. Let $\mathcal{A}$ be the set of such α (including X_0 itself), and let $\mathcal{L}$ be the set of those intermediate strings that loop with probability 1.

Define a probabilistic regular grammar $\mathcal{G}^\star$ using a non-terminal X_α for each $\alpha \in \mathcal{A}$, in addition to a special non-terminal Λ. For each pair $\alpha_1, \alpha_2 \in \mathcal{A}$, if there is a production in $\mathcal{G}$ that would rewrite α_1 as α_2, add to $\mathcal{G}^\star$ a production $(X_{\alpha_1} \to X_{\alpha_2})$. Likewise, if $\mathcal{G}$ has α rewrite in one step to a string in $\mathcal{L}$, include a production $(X_\alpha \to \Lambda)$. Finally, add a production $(X_{a^{k_i}} \to a^{k_i})$ for each $a^{k_i} \in S_{\mathcal{G}}^{+}$.

$\mathcal{G}^\star$ clearly generates the same distribution on $S_{\mathcal{G}}^{+}$ as the original grammar $\mathcal{G}$. By Proposition 13 we know that every such value $\mathbb{P}_{\mathcal{G}}(a^{k_i}) = \mathbb{P}_{\mathcal{G}^\star}(a^{k_i})$ must be rational. □

Using Proposition 18 we can also show that PCFGs, as we have defined them, are not in general closed under conditioning (though see Nederhof and Satta 2003 for a proof that PCFGs are closed under conditioning with regular sets if we allow irrational weights):

Proposition 19. *PCFGs are not closed under conditioning, even with finite sets.*

Proof. Consider the distribution in Example 17 together with the finite set $\{a, aa\}$. Evidently, $\mathbb{P}(a \mid \{a, aa\}) = \frac{20-2\sqrt{2}}{21}$, which is not a rational number. Because this distribution has finite support, Proposition 18 shows it cannot be defined by a PCFG. □

5.1 Algebraic Generating Functions

The probabilities in Examples 13-17 are all algebraic numbers. This is an instance of a more general result about the pgf for any PCFG $\mathcal{G}$. Assume $\Sigma = \{a\}$ and $\mathcal{N} = \{X_0, \ldots, X_n\}$. We will define a polynomial equation $x_i = Q_i(z, x_0, \ldots, x_n)$ in $n+2$ variables for each non-terminal X_i. For a string $\alpha \in (\mathcal{N} \cup \Sigma)^*$ let $\widehat{\alpha}$ be the result of replacing a with z, and X_i with x_i for each non-terminal X_i (and let $\widehat{\alpha} = 1$ if $\alpha = \epsilon$). Suppose in $\mathcal{G}$ we have both $X_i \xrightarrow{1/2} \alpha_1$ and $X_i \xrightarrow{1/2} \alpha_2$. Then our equation for X_i is:

$$x_i \quad = \quad \frac{1}{2}\,\widehat{\alpha_1} + \frac{1}{2}\,\widehat{\alpha_2} \tag{6}$$

with concatenation interpreted as multiplication. Likewise, if $\mathcal{G}$ has $X_i \xrightarrow{1} \alpha$, then the equation for X_i is simply $x_i = \widehat{\alpha}$. $\mathcal{G}$ thus produces a system of $n+1$ polynomial equations:

$$\begin{pmatrix} x_0 \\ \vdots \\ x_n \end{pmatrix} \quad = \quad \begin{pmatrix} Q_0(z, x_0, \ldots, x_n) \\ \vdots \\ Q_n(z, x_0, \ldots, x_n) \end{pmatrix} \tag{7}$$

Example 18. Returning to Examples 3, 13, we obtain equation:

$$x \quad = \quad \frac{1}{2}x^2 z + \frac{1}{2}z$$

Solving for x gives an expression for the pgf, $\mathfrak{G}(z) = \left(1 - \sqrt{1-z^2}\right)/z$.

By Theorem 15, even though the distribution in Example 3 only involves rational probability values, we have (see Kornai 2008; Bhattiprolu et al. 2017 for related observations):

Proposition 20. *The symmetric random walk distribution from Examples 3 and 13 cannot be defined by any probabilistic regular grammar.*

Toward a similar general limitative result for PCFGs, denote by g_i the generating function $\mathfrak{G}_{\mathcal{G}}^{X_i}(z) = \sum_{k=0}^{\infty} \mathbb{P}_{\mathcal{G}}^{X_i}(a^k) z^k$ corresponding to the probability function $\mathbb{P}_{\mathcal{G}}^{X_i}$, and let $\mathbf{g}$ be the vector $(g_0, \ldots, g_n)^T$. Lemma 21 now follows by a routine verification. (We leave off the subscript $\mathcal{G}$ in what follows.)

Lemma 21. *The vector* $\mathbf{g}$ *is a solution to the system of equations in (7); in other words,* $g_i = Q_i(z, g_0, \ldots, g_n)$ *for all* $i \leq n$.

Proof. Suppose there is a single production for X_i, rewriting to a string with j occurrences of a and non-terminals $X_{j_0}, \ldots, X_{j_m}$ (with possible repetitions). First note that:

$$\mathbb{P}^{X_i}(a^{j+k}) \quad = \quad \sum_{(l_1+\cdots+l_m=k)} \prod_{t \leq m} \mathbb{P}^{X_{j_t}}(a^{l_t})$$

This implies that

$$\begin{aligned}
\mathfrak{G}^{X_i}(z) &= \sum_{k=0}^{\infty} \mathbb{P}^{X_i}(a^{j+k}) z^{j+k} \\
&= \sum_{k=0}^{\infty} \Big(\sum_{(l_1+\cdots+l_m=k)} \prod_{t \leq m} \mathbb{P}^{X_{j_t}}(a^{l_t}) \Big) z^{j+k} \\
&= z^j \sum_{k=0}^{\infty} \Big(\sum_{(l_1+\cdots+l_m=k)} \prod_{t \leq m} \mathbb{P}^{X_{j_t}}(a^{l_t}) \Big) z^k \\
&= z^j \prod_{t \leq m} \Big(\sum_{k=0}^{\infty} \mathbb{P}^{X_{j_t}}(a^k) z^k \Big) \\
&= z^j \prod_{t \leq m} \mathfrak{G}^{X_{j_t}}(z)
\end{aligned}$$

But $z^j x_{j_0} \ldots x_{j_m}$ is the exactly the monomial Q_i. Similarly, if X_i has two productions, each with probability $1/2$, a very similar argument goes through by appeal to Eq. 6. □

In particular, the first component $g_0 = \mathfrak{G}_{\mathcal{G}}(z)$ is the pgf for $\mathcal{G}$ overall. Lemma 21 does not quite show that g_0 is an algebraic function in the sense of §2.2. We need to show that there is a polynomial equation $0 = Q(y, z)$ in variables y and z such that $y = g_0$ is a solution. However, it is known from elimination theory that for any system like (7) there is such a polynomial $Q(y, z)$ such that the first component of any solution to (7) is also a root of $Q(y, z)$. See the Elimination Theorem on p. 16 of Cox et al. (2000). (See also the closely related discussions in Kuich and Salomaa 1986; Flajolet and Sedgewick 2001; Panholzer 2005.) Thus, we have the following result (cf. Booth and Thompson 1973; Yao 1985):

Theorem 22. *The probability generating function for any PCFG on $\Sigma = \{a\}$ is algebraic.*

In particular this means that there is no PCFG encoding a Poisson distribution (Example 1). The fact that PCFGs cannot represent all algebraic pgfs of course follows from Proposition 18 (cf. Proposition 24 below).

Note that Theorem 22 is importantly different from the classic Chomsky-Schützenberger Theorem for non-probabilistic grammars (see Chomsky and Schützenberger 1963; Kuich and Salomaa 1986). The latter concerns word length counts, and states that the associated generating functions are algebraic for *unambiguous* context-free grammars. Indeed, the fact that context-free languages may possess transcendental generating functions is a useful tool for showing inherent ambiguity of a context-free language (Flajolet, 1987). By contrast, Theorem 22 applies to all PCFGs.

At the same time, it has been shown that ambiguity is necessary for PCFGs to go beyond the power of PRGs for a one-letter alphabet: any distribution represented by a PCFG that is merely *polynomially ambiguous*—meaning that the number of parses for each word is bounded by a fixed polynomial in the word's length—can be represented by a PRG (Bhattiprolu et al., 2017), and hence will possess a rational probability generating function. It of course follows that the distributions in Examples 13-17 are all exponentially ambiguous.

Finally, observe that essentially the same method used here to prove Theorem 22 could provide an alternative method of proof for Theorem 15, since for a PRG the system in (7) would be one of linear equations, and Gaussian elimination would produce a polynomial equation of degree 1.

6 Probabilistic Indexed Grammars

Many grammar formalisms in between context-free and context-sensitive have been explored in computational linguistics (see, e.g., Kallmeyer 2010 for an overview covering many of them, including probabilistic extensions), and some of these have featured in psychological models of language processing (e.g., Levy 2008; Nelson et al. 2017). Yet there has been almost no study of their expressive capacity for defining probability distributions.

Among the most well studied in the non-probabilistic setting are the *indexed grammars*, due to Aho (1968). The presentation here is a variation on that in Hopcroft and Ullman (1979). To $\mathcal{N}$ and Σ we add a finite set $\mathcal{I}$ of *indices*. Non-terminals are associated with a stack of indices and can pass on this stack to other non-terminals, in addition to pushing and popping indices from the stack at each step. Productions apply now to a pair $X[l]$ of non-terminal together with a topmost index $l \in \mathcal{I} \cup \{\epsilon\}$ (which may be empty) on its stack. Productions may be of three types, where the third is relevant only when $l \neq \epsilon$:

1. $X[l] \to \alpha[l]$ (copy indices to all non-terminals in α)
2. $X[l] \to \alpha[kl]$ (push index k and copy the result)
3. $X[l] \to \alpha$ (pop index l and copy the rest)

In a parse, each non-terminal is tagged with a stack of indices, with the start symbol X_0 initially carrying the empty index stack. The stack attached to the non-terminal on the left is then copied to all of the non-terminals appearing on the right-hand-side of the production, modulo a pop or push, while elements of Σ do not carry index stacks.

As in §2.3, a *probabilistic indexed grammar* satisfies the restriction that each pair of a non-terminal X and a top index $l \in \mathcal{I} \cup \{\epsilon\}$ appears on the left of 0, 1, or 2 productions, again with the usual probabilistic interpretation. All of the other definitions from §2.3 similarly remain unchanged. The main point about these grammars is the following:

Proposition 23. *Probabilistic indexed grammars can define distributions with transcendental pgfs.*

Proof. Here is a grammar defining $\mathbb{P}(a^{2^k}) = 2^{-k}$:

$$X_0[] \xrightarrow{1} Y[l] \qquad Y[l] \xrightarrow{1/2} Y[ll] \qquad Y[l] \xrightarrow{1/2} Z[l] \qquad Z[l] \xrightarrow{1} ZZ \qquad Z[] \xrightarrow{1} a$$

We observed in Lemma 1 that its pgf is transcendental. □

This shows that we have gone beyond the expressive capacity of PCFGs, even for a one-letter alphabet. There are still relatively simple distributions that probabilistic indexed grammars cannot define. For example, the factorial language $\{a^{k!} : k > 0\}$, while context-sensitive, has been shown beyond the capability of indexed grammars (Hayashi, 1973). *A fortiori* no probabilistic indexed grammar can define a distribution with this set as its support. Before moving to probabilistic context-sensitive grammars we first consider a natural and important restriction.

6.1 Probabilistic (Right-)Linear Indexed Grammars

A notable proper subclass of the indexed grammars are the *linear indexed grammars* (Duske and Parchmann, 1984; Gazdar, 1988), where we assume—similar to a regular grammar (§2.3)—that at most one non-terminal appears on the right-hand-side of a production. This is an example of a *mildly context-sensitive* formalism, weakly equivalent to other well studied grammatical formalisms like tree-adjoining grammar and combinatory categorial grammar (Joshi et al., 1991). Despite the restriction, it is straightforward to show that the these grammars also extend PCFGs (see Duske and Parchmann 1984 for the non-probabilistic case). That they do so properly is the next result.

Proposition 24. *Probabilistic linear indexed grammars can represent distributions that cannot be expressed by PCFGs.*

Proof. The following grammar defines a distribution with finite support, but also with irrational probabilities.

$$\begin{array}{lll} X_0[] \xrightarrow{1/2} a & Y[l] \xrightarrow{1/4} Y[ll] & Y[l] \xrightarrow{1/2} Y \\ X_0[] \xrightarrow{1/2} aY[l] & Y[l] \xrightarrow{1/4} a & Y[] \xrightarrow{1} \epsilon \end{array}$$

With $1/2$ probability X_0 rewrites to $aY[l]$, while $Y[l]$ in turn rewrites to ϵ with irrational probability $2 - \sqrt{2}$ (recall Examples 14 and 17). Thus, $\mathbb{P}(a) = \frac{3-\sqrt{2}}{2}$, while $\mathbb{P}(aa) = \frac{\sqrt{2}-1}{2}$. By Proposition 18, $\mathbb{P}$ cannot be defined by any PCFG. □

In fact, this grammar is not only linear, it is even *right-linear*, meaning that any non-terminal on the right-hand-side appears to the right of all terminal symbols (as in our definition of regular grammars from §2.3—in that case, right-linearity is not a substantive restriction; see Hopcroft and Ullman 1979). In the non-probabilistic setting, such grammars define exactly the context-free languages (Aho, 1968). But in the probabilistic setting they are evidently more expressive. As another example of a probabilistic right-linear indexed grammar, recall the tortoise and hare program defined earlier in Example 8:

Example 19 (Tortoise & Hare)**.** We can mimic this program as follows:

$$X_0[] \xrightarrow{1} Y[l] \qquad Y[l] \xrightarrow{1/6} aY[l] \qquad Y[l] \xrightarrow{1/6} aZ \qquad Y[] \xrightarrow{1} \epsilon$$
$$Y[l] \xrightarrow{1/2} aY[ll] \qquad Y[l] \xrightarrow{1/6} aY \qquad Z[l] \xrightarrow{1} Y \qquad Z[] \xrightarrow{1} \epsilon$$

The probability that $X_0[]$ rewrites to a^k is exactly the probability that the program in Example 8 returns k, i.e., that the hare catches up in k steps. Although the distribution is rational-valued, we conjecture that it is not PCFG-definable.

Given aforementioned results from the literature, Proposition 24 may seem surprising. Right-linear indexed grammars can be seen as grammatical versions of counter or pushdown automata (Duske et al., 1992), and there are several proofs implying that probabilistic pushdown automata and PCFGs are equally expressive (Abney et al. 1999; Chi 1999; see also the discussion in Smith and Johnson 2007). The apparent tension is resolved by observing that the proofs in Abney et al. (1999) and Chi (1999) require solutions to nonlinear equations in order to define the requisite probabilities in the corresponding equivalent PCFG. Proposition 18 simply shows that this cannot always be done if we only have recourse to rational weights. The correspondence thus breaks down in our setting. Probabilistic right-linear indexed grammars may therefore provide a better match to probabilistic pushdown automata or the equivalent ("multi-exit") recursive Markov chains.

Linear indexed grammars are unable to define the language $\{a^{2^k} : k > 0\}$ (they define only semi-linear languages in the sense of Parikh 1966; see Duske and Parchmann 1984); *a fortiori* probabilistic (right-)linear indexed grammars cannot define the distribution $\mathbb{P}(a^{2^k}) = 2^{-k}$, and thus fall short of general indexed grammars. In fact, following from their correspondence with pushdown automata (Duske et al., 1992), and drawing on existing results for the latter (Kuich and Salomaa, 1986, Theorem 14.15), these grammars can also be associated with algebraic generating functions.

7 Probabilistic Context-Sensitive Grammars

Grammars applied to natural language have typically been less powerful than the indexed grammars, and certainly less powerful than context-sensitive. In fact, over the decades there have been numerous explicit arguments that context-sensitive grammars are too complex (Chomsky, 1959; Savitch, 1987; Joshi et al., 1991). Probabilistic context-sensitive grammars (PCSGs) have rarely been considered, although they have occasionally appeared, e.g., in vision (Zhu and Mumford, 2007) and planning (Pynadath and Wellman, 1998). Propositions 29 and 30 below, together offer some further explanation for this relative absence.

In the classical hierarchy, context-sensitive grammars properly extend the indexed grammars (Aho, 1968), even in the case of a one-letter alphabet, witness the factorial language $\{a^{k!} : k > 0\}$ (Hayashi, 1973). Correspondingly, there are distributions definable by PCSGs that elude any probabilistic indexed grammar, as shown below in Proposition 28. To gain a better understanding of PCSGs, a helpful first observation is that, as in the classical case (Hopcroft and Ullman, 1979), the requirement on PCSGs given in §2.3 is equivalent to the intuitive stipulation that $|\alpha| \leq |\beta|$ whenever $(\alpha \to \beta) \in \Pi$. Such a *non-contracting* grammar may also include $X_0 \to \epsilon$ if X_0 does not appear on the right-hand-side of any production.

Lemma 25. *Any distribution defined by a non-contracting probabilistic grammar $\mathcal{G}_1$ can also be defined by a PCSG $\mathcal{G}_0$.*

Proof. For each $a \in \Sigma$ include a new non-terminal X_a and add production $X_a \to a$ to $\mathcal{G}_0$. Replace a throughout $\mathcal{G}_1$ with X_a. For each production in $\mathcal{G}_1$ of the form $(Y_1 \dots Y_n \to Z_1 \dots Z_m)$ with $2 \leq n \leq m$, add n non-terminals $W_1, \dots, W_n$, and $2n$ productions to $\mathcal{G}_0$:

$$\begin{aligned}
Y_1 Y_2 \dots Y_{n-1} Y_n &\to W_1 Y_2 \dots Y_{n-1} Y_n \\
W_1 Y_2 \dots Y_{n-1} Y_n &\to W_1 W_2 \dots Y_{n-1} Y_n \\
&\vdots \\
W_1 \dots W_{n-1} Y_n &\to W_1 \dots W_{n-1} W_n Z_{n+1} \dots Z_m \\
W_1 \dots W_n Z_{n+1} \dots Z_m &\to Z_1 \dots W_{n-1} W_n Z_{n+1} \dots Z_m \\
&\vdots \\
Z_1 \dots W_n Z_{n+1} \dots Z_m &\to Z_1 \dots Z_n Z_{n+1} \dots Z_m
\end{aligned}$$

The PCSG $\mathcal{G}_0$ thus defines the same distribution as $\mathcal{G}_1$. □

Lemma 25 suggests a natural class of PTMs corresponding to PCSGs, namely those that never write a blank symbol $\sqcup$, so called *non-erasing PTMs*. The non-probabilistic version of these machines was studied early on by Wang (1957) (see also Minsky 1967), though the connection to context-sensitive grammars seems not to have been noted in the literature.

In this section we allow using one more symbol $\triangleleft$, and assume the convention for a PTM to output σ is having $\triangleright\sigma\triangleleft$ on the tape followed by an infinite sequence of blank symbols. It is evident that nothing from §3 will be affected by this.

Theorem 26. *PCSGs and non-erasing PTMs are equivalent.*

Proof. The proof of Proposition 2 nearly already shows that any non-erasing PTM can be emulated by a PCSG. Since a non-erasing PTM will never write $\sqcup$, we require only a slight modification of the construction so that X_{n+1} moves to the right of the string and finally rewrites to $\triangleleft$. All rules in this modified construction are non-contracting.

For the other direction, we need to show that a non-erasing PTM can identify the highest priority substring, flip a coin to determine whether to rewrite it and what to rewrite it as, and then finally to rewrite it, possibly moving the remainder of the string to the right in case the rewrite is longer. Identifying the highest priority substring α can be done without changing the string at all, provided we are always writing $\triangleleft$ at the end. When a rule $(\alpha \to \beta)$ is identified the machine enters a fixed subroutine which involves going to the end of α and checking whether that is also the end of the string. If it is, we move $\triangleleft$ over $|\beta| - |\alpha|$ places and then easily perform the fixed rewrite. If it is not the end, then we first replace the last symbol of α with another $\triangleleft$; then we start shifting all of the extra symbols to the right $|\beta| - |\alpha|$ places until we hit $\triangleleft$ to the left. At that point we know we are $|\alpha|$ places to the right of the start of α, so we return to the beginning of α and simply write out β, finally returning to the beginning of the string. □

Proposition 27. *PCSGs can define transcendental pgfs.*

Proof. There is a PCSG defining the distribution $\mathbb{P}(a^{2^k}) = 2^{-k}$, which we know by Lemma 1 has a transcendental pgf. This could be shown in two ways. First, we could explicitly define the PCSG, for instance by massaging the grammar in Example 7 into context-sensitive form (cf. Example 9.5 in Hopcroft and Ullman 1979). Alternatively, we can describe a non-erasing PTM that defines $\mathbb{P}(a^{2^k}) = 2^{-k}$ and appeal to Theorem 26. The machine first writes

aa. From this point it iteratively flips a coin and every time we see heads we double the string. This latter operation can be done without ever writing $\sqcup$: rewrite the first a as $\triangleleft$ and then move to the right until seeing $\sqcup$, rewriting it too as $\triangleleft$. Write a to the right of $\triangleleft$ and then go back until seeing the first a. If the next symbol to the left of that a is also an a, then repeat, rewriting it as $\triangleleft$ and moving to the end of the string writing another a. But if the next symbol to the left of the a is $\triangleleft$, then we know we are done, in which case we rewrite each $\triangleleft$ as a and move back to the $\triangleright$ symbol. Enter $s_\#$ upon seeing a first tails. □

Invoking Theorem 26 again, it is also straightforward to show that a Turing machine can copy a sequence of 1s an increasing number of times, continuing again until witnessing a tails, without ever erasing any symbols. This means that there is a PCSG defining the measure $\mathbb{P}(a^{k!}) = 2^{-k}$ for $k > 0$, which implies:

Proposition 28. *PCSGs can define distributions that elude probabilistic indexed grammars.*

PCSGs can thus define complex distributions. The following proposition gives a further sense of how complex they can be. The statement is similar to an earlier result on non-probabilistic context-sensitive grammars due to Savitch (1987):

Proposition 29. *Consider any computably enumerable semi-measure $\mathbb{P} : \Sigma^* \to [0, 1]$. There is a PCSG (equivalently, non-erasing PTM) on augmented vocabulary $\Sigma \cup \{\triangleleft\}$ defining a semi-measure $\tilde{\mathbb{P}}$ such that $\mathbb{P}(\sigma) = \sum_n \tilde{\mathbb{P}}(\sigma\triangleleft^n)$ for all $\sigma \in \Sigma^*$.*

Proof Sketch. In the construction of a probabilistic grammar from a (possibly erasing) PTM in the proof of Theorem 2, whenever the PTM would write a $\sqcup$, instead write a $\triangleleft$. □

That is, the probability that a PTM returns σ is exactly the same as the probability that the PCSG returns σ together with some extra "dummy" symbols tacked on to the end. Despite the fact that PCSGs in some sense encompass all of the complexity of enumerable semi-measures, there is another sense in which they are even weaker than PCFGs.

Proposition 30. *For every PCSG $\mathcal{F}$ and every word σ, the probability $\mathbb{P}_\mathcal{G}(\sigma)$ is a rational number.*

Proof. We again invoke Theorem 26 to prove the result. Suppose we are given a non-erasing PTM $\mathcal{T}$. We show that $\mathbb{P}_\mathcal{T}(\sigma)$ is rational for every σ. The critical observation is that any non-erasing PTM $\mathcal{T}$ that produces a string $\triangleright\sigma\triangleleft$ will only ever write to $|\sigma| + 2$ tape cells, on any possible successful execution. So we can consider the finite set $\mathcal{C}$ of all possible configurations $[s, \eta, i]$ leading up to $\triangleright\sigma\triangleleft$ being on the tape in state $s_\#$ while reading $\triangleright$. Here, s is the current state, η is the string written on the $|\eta| = |\sigma| + 2$ many tape cells, and $i < |\sigma| + 2$ is the index of the symbol currently being read.

From this description we can naturally define a PRG over alphabet $\{a\}$ as follows. Include a non-terminal X_c for each configuration $c \in \mathcal{C}$ other than the final one, and let X_0 correspond to the initial configuration $[s_0, \triangleright\sqcup^{|\sigma|+1}, 0]$. If configuration c_2 follows configuration c_1 when $\mathcal{T}$ is reading random bit 0, then—unless c_2 is the final configuration—include a rule $X_{c_1} \to X_{c_2}$; and likewise when $\mathcal{T}$ is reading random bit 1. If in either case c_2 is the final configuration, then include a rule $X_{c_1} \to a$. Evidently, $\mathbb{P}_\mathcal{T}(\sigma) = \mathbb{P}_\mathcal{G}(a)$, and so the result follows from Proposition 13. □

Proposition 30 may seem perplexing in light of the fact that the context-sensitive languages properly extend the context-free languages, and even the indexed languages. The crux of the matter is the treatment of the empty string. Going back to Chomsky (1959),

the context-sensitive languages only extend the context-free provided we stipulate that we can always add the empty string to a language, e.g., by allowing $(X_0 \to \epsilon)$ when X_0 never occurs on the right-hand-side of any production. Examples 14-17 from §5 and Proposition 30 together show that this maneuver reverberates in a serious way once we turn to the probabilistic setting. Perhaps unintuitively, PCSGs cannot even define the seemingly simple distributions in Examples 14 and 17, for instance. Thus, while it may be that "almost any language one can think of is context-sensitive" (Hopcroft and Ullman, 1979), this is evidently not true of distributions defined by probabilistic context-sensitive grammars.

From Corollary 9 and Propositions 13, 18, and 30 we obtain:

Corollary 31. *PRGs, PCFGs, and PCSGs all define exactly the same finite-support distributions, namely the rational-valued finite-support distributions.*

By this point we have substantiated all of the qualitative relationships in the probabilistic hierarchy depicted in Figure 1. The fact that there are distributions defined by probabilistic linear indexed grammars but by neither PCSGs nor PCFGs follows from Propositions 24 and 30. Perhaps less obviously, we also now know that there are distributions defined by probabilistic indexed grammars that elude both their linear variant and PCSGs. To see this, let $\mathcal{G}_4$ be a probabilistic indexed grammar with $\mathbb{P}_{\mathcal{G}_1}(a^{2^k}) = 2^{-k}$ for $k > 1$ (easily definable by a variation on the construction in the proof of Proposition 23), and let $\mathcal{G}_5$ be the grammar presented in the proof of Proposition 24. Define a new grammar $\mathcal{G}_6$ employing a new start symbol that rewrites to the start symbols of $\mathcal{G}_4$ and $\mathcal{G}_5$, each with probability $1/2$. Evidently $\mathbb{P}_{\mathcal{G}_6}$ assigns probability $\frac{3-\sqrt{2}}{4}$ to a, probability $\frac{\sqrt{2}-1}{4}$ to aa, and probability $1/2^{k+1}$ to every string a^{2^k}, $k > 1$. Clearly $\mathbb{P}_{\mathcal{G}_6}$ cannot be defined using a linear indexed grammar or by any PCSG. Furthermore we require only a one-letter alphabet.

8 Conclusion and Further Issues

The hierarchy of expressible distributions that emerges in Figure 1 reveals how much the probabilistic setting can differ from the classical setting of formal language theory. Proper inclusions going up the hierarchy mirroring the classical case might have been expected; yet we reach incomparability already between the context-sensitive and context-free distributions, a pattern that continues through the indexed languages. Furthermore, grammar classes that are classically equivalent—such as the context-free and right-linear indexed grammars—come apart in the probabilistic setting. And unlike in the classical setting, all of these relationships can be observed even with distributions on one-letter alphabets, *pace* Parikh's theorem. A third theme is the significance of finite-support distributions. Probabilistic context-free grammars define algebraic probability generating functions and probabilistic context-sensitive grammars can define transcendental pgfs; yet when it comes to finite-support distributions they both collapse into the rational-valued measures, already definable by probabilistic regular grammars. Of the grammars studied here, only probabilistic (right-linear) indexed (and of course unrestricted) grammars can define irrational-valued measures with finite support.

8.1 Outstanding Mathematical Questions

Despite clarifying all of the inclusion/exclusion relationships among the classes in Figure 1, and establishing a partial correspondence between the probabilistic grammar hierarchy

and the analytical hierarchy (rational, algebraic, transcendental pgfs), a number of fundamental open questions remain. Some of the most significant technical questions include:

1. We have provided a full characterization of the distributions defined by probabilistic grammars and by almost-surely terminating grammars in general. But is it possible to provide full and informative characterizations any of the proper subclasses?

2. In particular, are the distributions defined by PRGs with rational weights exactly the rational-valued distributions with rational pgfs?

3. What (proper) subclass of algebraic pgfs do PCFGs define?

4. Are the distributions defined by probabilistic right-linear indexed grammars exactly the algebraic-number-valued distributions with algebraic pgfs? Or is there another class of grammars that would define this class of distributions?

5. Which subclasses of transcendental pgfs do PCSGs and indexed grammars define? Can we show that distributions like that in Example 8 cannot be defined by PCFGs?

6. What other closure results for conditioning can be obtained, aside from Theorems 6, 7, and 17, and Proposition 19?

7. Do we obtain more interesting distributions by going beyond the indexed languages to higher levels of the so called *hierarchy of indexed languages* (Maslov, 1976)?

8. How high in the hierarchy do we need to go in order to define a Poisson distribution (Example 1), or the distribution in Example 9, among others?

9. What other natural operations defined on grammars make sense in the context of probabilistic modeling? For instance, suppose we added a special primitive nonterminal P that produces a string a^k with probability given by a Poisson distribution? What would be obtain by adding this to the probabilistic regular grammars, or to other grammar classes?

10. Which distributions can be defined by adding probabilities in the so called subregular hierarchy, including classes that do not correspond to any natural grammar notions (Schützenberger, 1965; Jäger and Rogers, 2012)?

11. What can be said about classes between regular and context-free grammars, such as the linear or the "simple" context-free grammars (Hopcroft and Ullman, 1979)?

12. How might the results reported here shed light on questions about how tractably distributions at one level of the hierarchy can be *approximated* by distributions defined at lower levels? (See, e.g., Mohri and Nederhof 2000 for some results on this.)

To be sure, one can imagine many more natural questions still to be answered.

8.2 Generative Models in Psychology

Conceptually, what does the pattern of results reported here mean for psychological modeling? Questions about the tradeoff between expressivity and complexity, along various dimensions, of psychological models are quite familiar in the study of natural language, at

least since Chomsky (1959, 1965). But they are also of interest in nearly every other psychological domain. A guiding methodological principle is to give special attention to the simplest candidate models—those that require least of the agent being modeled—among those that adequately capture the phenomena of interest (see, e.g., Feldman 2016). Human learning is assumed to abide by a similar maxim, preferring the simplest hypotheses that account for observed patterns (Chater and Vitányi, 2003).

The (probabilistic) Chomsky-Schützenberger hierarchy highlights a salient notion of complexity, one that harmonizes well with other independent and potentially relevant notions of complexity (e.g., the computational cost of parsing; see Pratt-Hartmann 2010). Parts of the hierarchy have furthermore been used to explain differences in the processing of quantifier expressions in natural language (Szymanik and Zajenkowski, 2010), and have even been claimed to demarcate a meaningful boundary between human and non-human thought (e.g., Berwick et al. 2011; for a dissenting view see, for instance, Jackendoff 2011).

It is often asserted that we essentially know without any further ado that the human mind can at most be a (probabilistic) finite-state machine, if only because the brain itself is finite (see, e.g., Eliasmith 2010; Petersson et al. 2012 for different expressions of this view). On this picture the entire hierarchy collapses into the finite-state/regular. At the other extreme we see claims to the effect that the phenomena force us to the top of the hierarchy:

> Formalizing the full content of intuitive theories appears to require Turing-complete compositional representations, such as probabilistic first-order logic and probabilistic programming languages (Tenenbaum et al., 2011, p. 1284).

Our study lends support to both of these positions. On the one hand, Theorem 3 shows definitively that the class of all probabilistic grammars—which also express exactly the distributions defined by Turing-complete probabilistic programming languages—can encode virtually any discrete distribution that has ever been used for modeling, including those that strictly elude lower levels in the hierarchy. On the other hand, Corollary 9 shows that the much simpler probabilistic regular grammars can already define a number of prominent distributions, and are capable in principle of providing effective *approximations* to any distribution whatsoever; this is noteworthy insofar as approximation is inevitable when encoding continuous distributions anyway (recall the discussion in §4.1). Furthermore, both of these classes of distributions are closed under conditioning with the corresponding natural classes of events (Theorems 7 and 17).

Needless to say, for many purposes neither extreme may be apt. Feasible finite-state distributions may provide too poor approximations for targets of interest, whereas the enumerable semi-measures—perhaps learnable in theory (Chater and Vitányi, 2003; Vitányi and Chater, 2017)—may form too large a class for tractable learning. In between the two extremes are natural classes capturing useful and intuitive distributions such as those associated with random walks (Examples 3, 8) or those associated with complex but constrained grammatical phenomena in natural language (Chomsky, 1965; Joshi et al., 1991).

A recent trend has been to construe much of learning, even outside the domain of language, as a matter of inducing stochastic procedures, formalized as grammars or programs (see, e.g., Lake et al. 2015). Here the tradeoff between expressivity and tractability becomes especially poignant: some bias toward simpler programs is highly desirable, but it should not be so strong that it frustrates discovery of a good enough probabilistic model. Results like those reported here help us understand one side of this essential tradeoff.

At the same time, from a skeptical viewpoint, the pattern of results may cast doubt on whether the framework overall is appropriate for capturing the most important computa-

tional distinctions in cognition. As mentioned at the outset, the instantaneous firing rate of cortical neurons is assumed to be approximately Poisson distributed. Thus, in some sense, the Poisson distribution appears at what is plausibly a very basic level of cortical computation. In sharp contrast, this distribution evades expression until only the top level of the probabilistic Chomsky-Schützenberger hierarchy. The source of this discrepancy is not the restriction to rational parameters either, for instance, to be addressed by adding $e^{-\lambda}$ as a production weight; rather, the issue is structural (Theorem 15, Proposition 22).

To be sure, there are a variety of computational frameworks founded on more brain-like temporal and network dynamics (e.g., Maass et al. 2002, among many others). Such frameworks may ultimately carve the space of computational models along different dimensions. However, even within these research programs some of the most pressing questions concern how such frameworks might plausibly encode probabilistic generative models (Buesing et al., 2011) and other combinatorial generative mechanisms (Dehaene et al., 2015; Nelson et al., 2017). For much of psychology and cognitive science, theorizing in terms of generative models is intuitive, simple, and productive. We would like a comprehensive theoretical understanding of the capabilities and limitations of these models.

Acknowledgements

Part of this work was supported by the Center for the Study of Language and Information, Stanford University. The author would like to thank audiences at the Chennai Institute of Mathematical Sciences, Indiana University, Stanford University, and the Center for Brains, Minds, and Machines at MIT. Thanks especially to the editors and reviewers at *Journal of Mathematical Psychology*, and to Johan van Benthem, Roger Levy, Alex Lew, Larry Moss, Milan Mosse, R. Ramanujam, and Josh Tenenbaum for helpful comments and questions.

References

Abbott, J. T., Austerweil, J. L., and Griffiths, T. L. (2015). Random walks on semantic networks can resemble optimal foraging. *Psychological Review*, 122(3):558–569.

Abney, S., McAllester, D. A., and Pereira, F. (1999). Relating probabilistic grammars and automata. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 542–549.

Ackerman, N. L., Freer, C. E., and Roy, D. M. (2019). On the computability of conditional probability. *Journal of the ACM*, 66(3):1–40.

Aho, A. (1968). Indexed grammars—an extension of context-free grammars. *Journal of the ACM*, 15(4):647–671.

Berwick, R. C., Okanoya, K., Backers, G. J., and Bolhuis, J. J. (2011). Songs to syntax: the linguistics of birdsong. *Trends in Cognitive Sciences*, 15(3):113–121.

Bhattiprolu, V., Gordon, S., and Viswanathan, M. (2017). Parikh's Theorem for weighted and probabilistic context-free grammars. In Bertrand, N. and Bortolussi, L., editors, *Quantitative Evaluation of Systems (QEST 2017)*. Lecture Notes in Computer Science, vol 10503. Springer.

Billingsley, P. (1999). *Convergence of Probability Measures*. Wiley Series in Probability and Mathematical Statistics, 2 edition.

Booth, T. L. and Thompson, R. A. (1973). Applying probability measures to abstract languages. *IEEE Transactions on Computers*, C-22(5):442–450.

Buesing, L., Bill, J., Nessler, B., and Maass, W. (2011). Neural dynamics as sampling: A model for stochastic computation in recurrent networks of spiking neurons. *PLoS Computational Biology*, 7(11).

Busemeyer, J. R. and Diederich, A. (2002). Survey of decision field theory. *Mathematical Social Sciences*, 43:345–370.

Chaitin, G. J. (1975). A theory of program size formally identical to information theory. *Journal of the ACM*, 22(3):329–340.

Chater, N. and Manning, C. D. (2006). Probabilistic models of language processing and acquisition. *Trends in Cognitive Sciences*, 10(7):335–344.

Chater, N. and Oaksford, M. (2013). Programs as causal models: Speculations on mental programs and mental representation. *Cognitive Science*, 37(6):1171–1191.

Chater, N. and Vitányi, P. (2003). Simplicity: a unifying principle in cognitive science? *Trends in Cognitive Sciences*, 7(1):19–22.

Chi, Z. (1999). Statistical properties of probabilistic context-free grammars. *Computational Linguistics*, 25(1):131–160.

Chomsky, N. (1959). On certain formal properties of grammars. *Information & Control*, 2:137–167.

Chomsky, N. (1965). *Aspects of the Theory of Syntax*. MIT Press.

Chomsky, N. and Schützenberger, M. P. (1963). The algebraic theory of context-free languages. In Hirschberg, D. and Braffort, P., editors, *Computer Programming and Formal Systems*, pages 118–161. North-Holland.

Cox, D., Little, J., and O'Shea, D. (2000). *Ideals, Varieties, and Algorithms*. Springer Verlag.

Dal Lago, H. and Zorzi, M. (2012). Probabilistic operational semantics for the lambda calculus. *RAIRO - Theoretical Informatics and Applications*, 46(3):413–450.

Dehaene, S., Meyniel, F., Wacongne, C., Wang, L., and Pallier, C. (2015). The neural representation of sequences: from transition probabilities to algebraic patterns and linguistic trees. *Neuron*, 88:2–19.

Denison, S., Bonawitz, E., Gopnik, A., and Griffiths, T. L. (2013). Rational variability in children's causal inferences: the sampling hypothesis. *Cognition*, 126:285–300.

Droste, M., Kuich, W., and Vogler, H., editors (2009). *Handbook of Weighted Automata*. Springer.

Dupont, P., Denis, F., and Esposito, Y. (2005). Links between probabilistic automata and hidden Markov models: probability distributions, learning models and induction algorithms. *Pattern Recognition*, 38:1349–1371.

Duske, J., Middendorf, M., and Parchmann, R. (1992). Indexed counter languages. *Theoretical Informatics & Applications*, 26(1):93–113.

Duske, J. and Parchmann, R. (1984). Linear indexed languages. *Theoretical Computer Science*, 32:47–60.

Eilenberg, S. (1974). *Automata, Language, and Machines*. Academic Press.

Eliasmith, C. (2010). How we ought to describe computation in the brain. *Studies in History & Philosophy of Science Part A*, 41(3):313–320.

Etessami, K. and Yannakakis, M. (2009). Recursive Markov chains, stochastic grammars, and monotone systems of nonlinear equations. *Journal of the ACM*, 65(1):1–66.

Faisal, A. A., Selen, L. P. J., and Wolpert, D. M. (2008). Noise in the nervous system. *Nature Reviews Neuroscience*, 9:292–303.

Feldman, J. (2016). The simplicity principle in perception and cognition. *Wiley Interdisciplinary Reviews: Cognitive Science*, 7(5):330–340.

Flajolet, P. (1987). Analytic models and ambiguity of context-free languages. *Theoretical Computer Science*, 49:283–309.

Flajolet, P., Pelletier, M., and Soria, M. (2011). On Buffon machines and numbers. In *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2011)*, pages 172–183.

Flajolet, P. and Sedgewick, R. (2001). Analytic combinatorics: Functional equations, rational and algebraic functions. Technical Report RR4103, INRIA.

Freer, C. E., Roy, D. M., and Tenenbaum, J. B. (2014). Towards common-sense reasoning via conditional simulation: Legacies of Turing in artificial intelligence. In Downey, R., editor, *Turing's Legacy*. ASL Lecture Notes in Logic.

Gács, P. (2005). Uniform test of algorithmic randomness over a general space. *Theoretical Computer Science*, 2005:91–137.

Gazdar, G. (1988). Applicability of indexed grammars to natural languages. In Reyle, U. and Rohrer, C., editors, *Natural Language Parsing and Linguistic Theories*. Springer Verlag.

Gershman, S. J. and Blei, D. M. (2012). A tutorial on Bayesian nonparametric models. *Journal of Mathematical Psychology*, 56(1):1–12.

Glimcher, P. W. (2005). Indeterminacy in brain and behavior. *Annual Review of Psychology*, 56:25–56.

Goodman, N. D. and Tenenbaum, J. B. (2016). Probabilistic Models of Cognition. `http://probmods.org/v2`. Accessed: 2019-8-5.

Griffiths, T. L., Daniels, D., Austerweil, J. L., and Tenenbaum, J. B. (2018). Subjective randomness as statistical inference. *Cognitive Psychology*, 103:85–109.

Griffiths, T. L., Steyvers, M., and Tenenbaum, J. B. (2007). Topics in semantic representation. *Psychological Review*, 114(2):211–244.

Harris, T. E. (1963). *The Theory of Branching Processes*. Springer.

Hayashi, T. (1973). On derivation trees of indexed grammars. *Publications of the Research Institute for Mathematical Sciences*, 9(1):61–92.

Hinton, G. E. and Sejnowski, T. J. (1983). Optimal perceptual inference. In *Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition*, pages 448–453.

Hopcroft, J. E. and Ullman, J. D. (1979). *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, 1st edition.

Icard, T. (2019). Why be random? *Mind*. DOI: 10.1093/mind/fzz065.

Icard, T. F. (2017a). Beyond almost-sure termination. In *Proceedings of the 39th Annual Meeting of the Cognitive Science Society*.

Icard, T. F. (2017b). From programs to causal models. In Cremers, A., van Gessel, T., and Roelofsen, F., editors, *Proceedings of the 21st Amsterdam Colloquium*, pages 35–44.

Jackendoff, R. (2011). What is the human language faculty? Two views. *Language*, 87:586–624.

Jäger, G. and Rogers, J. (2012). Formal language theory: Refining the Chomsky hierarchy. *Philosophical Transactions of the Royal Society B*, 367:1956–1970.

Johnson, M. (2010). PCFGs, topic models, adaptor grammars and learning topical collocations and the structure of proper names. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1148–1157.

Johnson, N. L., Kemp, A. W., and Kotz, S. (2005). *Univariate Discrete Distributions*. Wiley Series in Probability and Statistics, 3rd edition.

Joshi, A. K., Vijay-Shanker, K., and Weir, D. J. (1991). The convergence of mildly context-sensitive grammar formalisms. In Sells, P., Shieber, S. M., and Wasow, T., editors, *Foundational Issues in Natural Language Processing*. MIT Press.

Jungen, R. (1931). Sur les séries de Taylor n'ayant que des singularités algébrico-logarithmiques sur leur cercle de convergence. *Commentarii Mathematici Helvetici*, 3:226–306.

Kallmeyer, L. (2010). *Parsing Beyond Context-Free*. Springer Verlag.

Kaminski, B. L. and Katoen, J.-P. (2015). On the hardness of almost-sure termination. In *Mathematical Foundations of Computer Science*, pages 307–318.

Kim, Y., Dyer, C., and Rush, A. M. (2019). Compound probabilistic context-free grammars for grammar induction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2369–2385.

Klein, D. and Manning, C. D. (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Meeting of the Association for Computational Linguistics (ACL)*, pages 423–430.

Knill, D. C. and Pouget, A. (2004). The Bayesian brain: the role of uncertainty in neural coding and computation. *Trends in Neurosciences*, 27(12):712–719.

Knuth, D. E. and Yao, A. C. (1976). The complexity of nonuniform random number generation. In *Algorithms and Complexity*, pages 357–428. Academic Press.

Kornai, A. (2008). *Mathematical Linguistics*. Springer Verlag.

Kress, R. (1998). *Numerical Analysis*. Springer.

Kuich, W. and Salomaa, A. (1986). *Semirings, Automata, Languages*. Springer Verlag.

Lake, B. M., Salakhutdinov, R., and Tenenbaum, J. B. (2015). Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338.

de Leeuw, K., Moore, E. F., Shannon, C. E., and Shapiro, N. (1956). Computability by probabilistic machines. In *Automata Studies*, pages 183–212. Princeton University Press.

Levy, R. (2008). Expectation-based syntactic comprehension. *Cognition*, 106:1126–1177.

Li, Y., Schofield, E., and Gönen, M. (2019). A tutorial on Dirichlet process mixture modeling. *Journal of Mathematical Psychology*, 91:128–144.

Liang, P., Jordan, M. I., and Klein, D. (2010). Probabilistic grammars and hierarchical Dirichlet processes. In O'Hagan, T. and West, M., editors, *The Handbook of Applied Bayesian Analysis*, pages 776–822. Oxford University Press.

Lin, H. W. and Tegmark, M. (2017). Critical behavior in physics and probabilistic formal languages. *Entropy*, 19(299).

Maass, W., Natschläger, T., and Markram, H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 14:2531–2560.

MacKay, D. J. C. (1996). Equivalence of linear Boltzmann chains and hidden Markov models. *Neural Computation*, 8(1):178–181.

Maslov, A. N. (1976). Multilevel stack automata. *Problemy peredachi informatsii*, 12(1):55–62.

Miller, G. A. (1952). Finite Markov processes in psychology. *Psychometrika*, 17(2).

Minsky, M. (1967). *Computation: Finite and Infinite Machines*. Prentice Hall.

Mohri, M. and Nederhof, M.-J. (2000). Regular approximation of context-free grammars through transformation. In Junqua, J.-C. and van Noord, G., editors, *Robustness in Language and Speech Technology*, pages 252–261. Kluwer.

Nederhof, M.-J. and Satta, G. (2003). Probabilistic parsing as intersection. In *Proceedings of the 8th International Conference on Parsing Technologies*.

Nelson, M. J., El Karoui, I., Giber, K., Yang, X., Cohen, L., Koopman, H., Cash, S. S., Naccache, L., Hale, J. T., Pallier, C., and Dehaene, S. (2017). Neurophysiological dynamics of phrase-structure building during sentence processing. *Proceedings of the National Academy of Sciences*, 114(18):3669–3678.

Nishioka, K. (1996). *Mahler Functions and Transcendence*. Springer Verlag.

Nosofsky, R. M. and Palmeri, T. J. (1997). An exemplar-based random walk model of speeded classification. *Psychological Review*, 104(2):266–300.

O'Cinneide, C. A. (1990). Characterization of phase-type distributions. *Communications in Statistics. Stochastic Models*, 6:1–57.

Olmedo, F., Kaminski, B. L., Katoen, J.-P., and Matheja, C. (2016). Reasoning about recursive probabilistic programs. In *Proceedings of the 31st Annual IEEE Symposium on Logic in Computer Science (LICS)*, pages 672–681.

Orbán, G., Berkes, P., Fiser, J., and Lengyel, M. (2016). Neural variability and sampling-based probabilistic representations in the visual cortex. *Neuron*, 92:530–543.

Panholzer, A. (2005). Gröbner bases and the defining polynomial of a context-free grammar generating function. *Journal of Automata, Languages & Combinatorics*, 1:79–97.

Parikh, R. (1966). On context-free languages. *Journal of the ACM*, 13(4):570–581.

Paz, A. (1971). *Introduction to Probabilistic Automata*. Academic Press.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Petersson, K.-M., Folia, V., and Hagoort, P. (2012). What artificial grammar learning reveals about the neurobiology of syntax. *Brain & Language*, 120:83–95.

Petre, I. (1999). Parikh's theorem does not holds for multiplicities. *Journal of Automata, Languages, & Combinatorics*, 4(3):17–30.

Pratt-Hartmann, I. (2010). Computational complexity in natural language. In Clark, A., Fox, C., and Lappin, S., editors, *The Handbook of Computational Linguistics and Natural Language Processing*, pages 43–73. Blackwell.

Putnam, H. (1967). Psychological predicates. In Capitan, W. H. and Merrill, D. D., editors, *Art, Mind, and Religion*. Pittsburgh University Press.

Pynadath, D. V. and Wellman, M. P. (1998). Generalized queries on probabilistic context-free grammars. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 20(1):65–77.

Rabin, M. O. (1963). Probabilistic automata. *Information & Control*, 6(3):230–245.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, 85(2):59–108.

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

Sanborn, A. N., Griffiths, T. L., and Navarro, D. J. (2010). Rational approximations to rational models: Alternative algorithms for category learning. *Psychological Review*, 117(4):1144–1167.

Savitch, W. J. (1987). Context-sensitive grammar and natural language syntax. In Savitch, W. J., Bach, E., Marsh, W., and Safran-Naveh, G., editors, *The Formal Complexity of Natural Language*, pages 358–368. Springer (Studies in Linguistics and Philosophy).

Schützenberger, M. P. (1961). On the definition of a family of automata. *Information & Control*, 4:245–270.

Schützenberger, M. P. (1965). On finite monoids having only trivial subgroups. *Information & Control*, 8:190–194.

Smith, N. A. and Johnson, M. (2007). Weighted and probabilistic context-free grammars are equally expressive. *Computational Linguistics*, 33(4):477–491.

Soittola, M. (1976). Positive rational sequences. *Theoretical Computer Science*, 2:313–321.

Stanley, R. P. (2011). *Enumerative Combinatorics*, volume 1. Cambridge University Press.

Szymanik, J. and Zajenkowski, M. (2010). Comprehension of simple quantifiers: Empirical evaluation of a computational model. *Cognitive Science*, 34(3):521–532.

Tenenbaum, J. B., Griffiths, T. L., and Kemp, C. (2006). Theory-based Bayesian models of inductive learning and reasoning. *Trends in Cognitive Sciences*, 10(7):309–318.

Tenenbaum, J. T., Kemp, C., Griffiths, T., and Goodman, N. D. (2011). How to grow a mind: Statistics, structure, and abstraction. *Science*, 331:1279–1285.

Townsend, J. T. and Ashby, F. G. (1983). *The Stochastic Modeling of Elementary Psychological Processes*. Cambridge University Press.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59:433–460.

Vitányi, P. M. B. and Chater, N. (2017). Identification of probabilities. *Journal of Mathematical Psychology*, 76(A):13–24.

von Neumann, J. (1951). Various techniques used in connection with random digits. *Journal of Research of the National Bureau of Standards, Applied Mathematics Series*, 12:36–38.

Vul, E., Hanus, D., and Kanwisher, N. (2009). Attention as inference: Selection is probabilistic; responses are all-or-none samples. *Journal of Experimental Psychology: General*, 138(4):546–560.

Wang, H. (1957). A variant to Turing's theory of computing machines. *Journal of the ACM*, 4(1):63–92.

Wilson, R. C., Geana, A., White, J. M., Ludwig, E. A., and Cohen, J. D. (2014). Humans use directed and random exploration to solve the explore-exploit dilemma. *Journal of Experimental Psychology: General*, 143(6):2074–2081.

Wood, I., Sunehag, P., and Hutter, M. (2011). (Non-)equivalence of universal priors. In Dowe, D., editor, *Solomonoff 85th Memorial Conference*, pages 417–425. Springer LNCS.

Yao, A. C. (1985). Context-free grammars and random number generation. In Apostolico, A. and Galil, Z., editors, *Combinatorial Algorithms on Words*, volume 12, pages 357–361. Springer.

Zhu, S.-C. and Mumford, D. (2007). A stochastic grammar of images. *Foundations & Trends in Computer Graphics & Vision*, 2(4):259–362.

Zvonkin, A. K. and Levin, L. A. (1970). The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms. *Uspekhi Matematicheskikh Nauk*, 25(6):85–127.