
Learning Acyclic Probabilistic Circuits Using Test Paths

Dana Angluin¹ and James Aspnes^{1,*} and Jiang Chen^{2,†} and David Eisenstat³ and Lev Reyzin^{1,‡}

¹ Computer Science Department, Yale University
{angluin, aspnes}@cs.yale.edu, lev.reyzin@yale.edu

² Yahoo! Inc., 701 First Avenue, Sunnyvale, CA 94086
criver@gmail.com

³ eisenstatdavid@gmail.com

Abstract

We define a model of learning probabilistic acyclic circuits using value injection queries, in which an arbitrary subset of wires is set to fixed values, and the value on the single output wire is observed. We adapt the approach of using test paths from the Circuit Builder algorithm [AACW06] to show that there is a polynomial time algorithm that uses value injection queries to learn Boolean probabilistic circuits of constant fan-in and log depth. In the process, we discover that test paths fail utterly for circuits over alphabets of size greater than two and establish upper and lower bounds on the attenuation factor for general and transitively reduced Boolean probabilistic circuits of test paths versus general experiments. To overcome the limitations of test paths for non-Boolean alphabets, we introduce function injection queries, which allow the symbols on a wire to be mapped to other symbols rather than just to themselves or constants.

1 Introduction

Probabilistic networks are used as models in a variety of domains, for example, gene interaction networks, social networks and causal reasoning. In a binary model of gene interaction, the state of each gene is either active or inactive, and the state of each gene is determined as a function of the states of some number of other genes, its inputs. In a probabilistic variant of the model, the activation function specifies, for each possible combination of the states of the inputs, the probability that the gene will be active. In the independent cascade model of social networks, the state of each agent is active or inactive and for each pair (u, v) of agents, there is a probability that the activation of u will cause v to become active. Kempe, Kleinberg and Tardos study the problem of maximizing influence in this and related models of social networks [KKET03, KKT05]. In a Bayesian network there

is an acyclic directed graph and a joint probability distribution over the node values such that the joint distribution is simply the product of each of the marginal distributions for each node given the values of the parents (in-neighbors) of the node.

A fundamental question is how much we can infer about the properties and structure of such networks from observing and experimenting with their behaviors. Prior research suggests that there is no polynomial time algorithm to learn Boolean functions represented by acyclic circuits of constant fan-in and depth $O(\log n)$ when we can set only the inputs of the circuit and observe only the output [AK95]. In this paper we consider a different setting, **value injection queries**, in which we can fix the values on any subset of wires in the target circuit, but still only observe the output of the circuit.

The idea of value injection queries was inspired by models of gene suppression and gene overexpression in the study of gene interaction networks [AKMM98, ITK00] and was proposed in [AACW06]. They show that with value injection queries, acyclic deterministic circuits with constant-size alphabets, constant fan-in and depth $O(\log n)$ are learnable up to behavioral equivalence in polynomial time. To extend these results to analog circuits, Angluin et al. [AACR07] consider circuits with polynomial-size alphabets. Larger alphabets make the learning problem significantly harder, necessitating structural restrictions on the graphs of the circuits to achieve polynomial time learnability. They show that with value injection queries, acyclic deterministic circuits that are transitively reduced (or in general, have constant shortcut width) and have polynomial-size alphabets, constant fan-in and unbounded depth are learnable up to behavioral equivalence in polynomial time.

In this paper we investigate how well the above positive results can be extended to the case of acyclic probabilistic circuits. The key technique in the previous work has been the idea of a **test path** for an arbitrary wire w in the circuit. Informally speaking, a test path is a directed path of wires from w to the output wire in which each wire is an input of the next wire on the path, and the other (non-path) inputs of wires on the path are fixed to constant values, thus isolating the wires along the path from the rest of the circuit. Ideally, the choice of constant values is made in such a way as to maximize the effect on the output of the circuit of changing w from one value to another. A test path thus functions as a kind of “microscope” for viewing the effects of different values on the wire w . The primary focus of this paper is to un-

*Supported in part by NSF grant CNS-0435201.

†Supported in part by a research contract from Consolidated Edison.

‡This material is based upon work supported under a National Science Foundation Graduate Research Fellowship.

derstand the properties of test paths in probabilistic circuits, and the extent to which they can be used to give polynomial time algorithms for learning probabilistic acyclic circuits.

In Section 2 we formally define our model of acyclic probabilistic circuits, value injection queries and distribution injection queries, behavioral equivalence, and the learning problem that we consider. In Section 3 we establish some basic results about probabilistic circuits and value and distribution injection experiments. In Section 4 we review the test path lemma used in previous work and show that it fails utterly in probabilistic circuits with alphabet size greater than two. However, for Boolean probabilistic circuits, we show that the test path lemma holds with an attenuation factor that depends on the structure of the circuit. (Lemma 10 treats general acyclic circuits and Corollary 11 specializes the bound to transitively reduced circuits.) In Section 5 we apply the test path lemma in the Boolean case to adapt the Circuit Builder algorithm [AACW06] to find using value injection queries, with high probability, in time polynomial in n and $1/\varepsilon$, a circuit that is ε -behaviorally equivalent to a target acyclic Boolean probabilistic circuit of size n with constant fan-in and depth bounded by a constant times $\log n$. In Section 6, we consider lower bounds on the attenuation of paths; Lemma 15 shows that our bound is tight for transitively reduced circuits and Lemma 17 gives a lower bound for the case of general acyclic circuits. In Section 7 we introduce a stronger kind of query, a **function injection query**, and show that test paths with function injections overcome the limitations of test paths for circuits with alphabets of size greater than two.

2 Model

2.1 Probabilistic Circuits

We extend the circuit learning model studied in [AACR07, AACW06], to probabilistic gates. An unusual feature of this model is that circuits do not have distinguished inputs—since the learning algorithm seeks to predict the output behavior of value injection experiments that override the values on an arbitrary subset of wires, each wire is a potential input. Probabilistic circuits are closely related to Bayesian networks as well; we have chosen, however, to retain the conventions of the previous works.

A **probabilistic circuit** C of **size** $n \geq 1$ has n **wires**, of which one is the distinguished **output wire**. We call the set of C 's wires W , and these wires take values in a finite **alphabet** Σ with $|\Sigma| \geq 2$. If $\Sigma = \{0, 1\}$, then C is **Boolean**. The value on a wire is ordinarily determined by the output of an associated probabilistic gate, whose distribution is a function of the values on other wires.

Formally, an **value distribution** D is a probability distribution over Σ , that is, a map from Σ to the real interval $[0, 1]$ such that $\sum_{\sigma \in \Sigma} D(\sigma) = 1$. The **support** of D is the set of values $\sigma \in \Sigma$ such that $D(\sigma) > 0$, and when this set is a singleton $\{\sigma\}$ for some $\sigma \in \Sigma$, we say D is **deterministic**. For nonempty sets of values $S \subseteq \Sigma$, the **uniform distribution** $U(S)$ is the distribution such that $U(S)(\sigma) = |\sigma \in S|/|S|$.

A k -ary **probabilistic gate function** f maps each k -tuple $(\sigma_1, \dots, \sigma_k) \in \Sigma^k$ of values to a value distribution. When C is Boolean, we can specify f by a truth table giving the

expected value for each Boolean vector of inputs. A probabilistic gate function is **deterministic** if it maps k -tuples to deterministic value distributions only.

A **probabilistic gate** g of **fan-in** k pairs a k -ary probabilistic gate function f with a k -tuple $(w_1, \dots, w_k) \in W^k$ of **input wires**. g is **deterministic** if f is deterministic. When $k = 0$, the gate g has no inputs, and we can regard it as a value distribution, or, when C is Boolean, a biased coin flip.

A **probabilistic circuit** C maps wires to probabilistic gates. C is **deterministic** if all of its gates are deterministic. The **fan-in** of C is the maximum fan-in over C 's gates. The **circuit graph** of C has nodes W and a directed edge (w, u) if w is one of the input wires of the gate associated with u . It is important to distinguish between wires in the circuit and edges in the circuit graph. For example, if wire w is an input of wires u and v , then there will be two directed edges, (w, u) and (w, v) , in the circuit graph.

Wire u is **reachable** from wire w if there is a directed path from w to u in the circuit graph. A wire is **relevant** if the output wire is reachable from it. The **depth** of a wire w is the number of edges in the longest simple path from w to the output wire in the circuit graph. The **depth** of the circuit is maximum depth of any relevant wire. The circuit is **acyclic** if the circuit graph contains no directed cycles. The circuit is **transitively reduced** if its circuit graph is transitively reduced, that is, if it contains no edge (w, u) such that there is a directed path of length at least two from w to u . In this paper we assume all circuits are acyclic.

2.2 Experiments

In an experiment some wires are constrained to be particular symbols or value distributions and the other wires are left free. The behavior of a circuit consists of its responses to all possible experiments. For probabilistic circuits we consider both value injection experiments and distribution injection experiments.

A **distribution injection experiment** e is a function with domain W that maps each wire w to a special symbol $*$ or to a value distribution. A **value injection experiment** e is a distribution injection experiment for which every value distribution assigned is deterministic – that is, always generates the same symbol. To simplify notation, we think of a value injection experiment as a mapping from W to $(\Sigma \cup \{*\})$. If e is either kind of experiment, we say that e leaves w **free** if $e(w) = *$; otherwise we say that e **constrains** w to $e(w)$. If $e(w)$ is a single symbol, then we say e **fixes** w to $e(w)$.

We define a partial ordering \leq on the set containing $*$ and all value distributions D as follows: $D \leq *$ for every value distribution D , and for two value distributions, $D_1 \leq D_2$ if the support of D_1 is a subset of the support of D_2 . This ordering is extended to experiments on the same set of wires W as follows: $e_1 \leq e_2$ if for every $w \in W$, $e_1(w) \leq e_2(w)$. The intuitive meaning of $e_1 \leq e_2$ is that e_1 is at least as constraining as e_2 for every wire.

If e is any experiment, w is a wire, and a is $*$ or an element of Σ or a value distribution, then the experiment $e|_{w=a}$ is defined to be the experiment e' such that $e'(w) = a$ and $e'(u) = e(u)$ for all $u \in W$ such that $u \neq w$.

2.3 Behavior

Let C be a probabilistic circuit. Then a distribution injection experiment e determines a joint distribution over assignments of elements of Σ to all of the wires of the circuit, as follows. If wire w is constrained then w is randomly and independently assigned a value in Σ drawn according to the value distribution $e(w)$; in the case of a value injection experiment, this just assigns a fixed element of σ to w . If wire w is free, has probabilistic gate function f and its inputs u_1, \dots, u_k have been assigned the values $\sigma_1, \dots, \sigma_k$, then w is randomly and independently assigned a value from Σ according to the value distribution $f(\sigma_1, \dots, \sigma_k)$.

Constrained gates and gates of fan-in zero give the base cases for the above recursive definition, which assigns an element of Σ to every wire because the circuit is acyclic. Let $C(e, w)$ denote the (marginal) value distribution of the assignments of values to w for the above process. The **output distribution** of the circuit, denoted $C(e)$, is the distribution $C(e, z)$, where z is the output wire of the circuit. The **behavior** of a circuit C is the function that maps value injection experiments e to output distributions $C(e)$.

2.4 Example: C_1

We give an example of a simple Boolean probabilistic circuit, which we will also refer to later. The 2-input **averaging gate** $A(b_1, b_2)$ outputs 1 with probability $(b_1 + b_2)/2$. We define a circuit C_1 of 4 wires as follows: $w_4 = A(w_2, w_3)$, $w_3 = w_1$, $w_2 = w_1$, and $w_1 = U(\{0, 1\})$. The output wire is w_4 . C_1 is depicted in Figure 1.

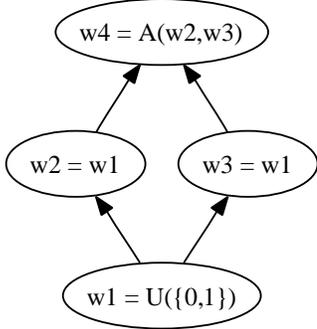


Figure 1: The circuit C_1 ; w_4 is the output wire.

To illustrate the behavior of this circuit, we consider two value injection experiments. Define the experiment e_1 to leave every wire in C_1 free, that is, $e_1(w_i) = *$ for $1 \leq i \leq 4$. Given e_1 , we construct one random outcome as follows. The wire w_1 is assigned a value as the result of an unbiased coin flip – say it is assigned 0. Then the values assigned to w_2 and w_3 are determined because they are each the output of an identity gate with w_1 as input: both are 0. Finally, because both its input wires have been assigned values, w_4 can be assigned a value according to $A(0, 0)$, which is deterministically 0. It is easy to see that this is one of two possible outcomes for experiment e_1 ; either all wires are assigned 0 or all wires are assigned 1, and these each occur with probability $1/2$. The output distribution $C_1(e_1)$ is just an unbiased coin flip.

Now consider experiment $e_2 = e_1|_{w_2=1}$ that fixes w_2 to 1 and leaves the other wires free. Once again, the value of w_1 is determined by a coin flip – say it is assigned 0. Since w_2 is fixed to 1, that is its assignment. Wire w_3 is free, and is therefore assigned the value of w_1 , that is 0. Now the inputs of w_4 have been assigned values, so we consider $A(1, 0)$, which randomly and equiprobably selects 0 or 1. If, instead, the coin flip for w_1 had returned 1, all wires would be assigned 1. There are three possible assignments to (w_1, w_2, w_3, w_4) for experiment e_2 : $(1, 1, 1, 1)$ with probability $1/2$, $(0, 1, 0, 0)$ with probability $1/4$ and $(0, 1, 0, 1)$ with probability $1/4$. The output distribution $C_1(e_2)$ is a biased coin flip that is 1 with probability $3/4$.

2.5 Behavioral Equivalence

Two circuits C and C' are **behaviorally equivalent** if they have the same set of wires, the same output wire and the same behavior, that is, for every value injection experiment e , $C(e) = C'(e)$. We also need a concept of approximate equivalence. The **(statistical) distance** between value distributions D and D' is $d(D, D') = (1/2) \sum_{\sigma} |D(\sigma) - D'(\sigma)|$, which takes values in $[0, 1]$. Note that when D and D' are deterministic, $d(D, D')$ is 0 if $D = D'$ and 1 otherwise. For $\varepsilon \geq 0$, C is ε -**behaviorally equivalent** to C' if they contain the same wires and the same output wire, and for every value injection experiment e , $d(C(e), C'(e)) \leq \varepsilon$, where d is the distance between value distributions defined above.

In Lemma 2 we show that the behavioral equivalence of C and C' implies $C(e) = C'(e)$ for all distribution injection experiments as well. Note that even when all the gates are Boolean, deterministic and relevant, the circuit graph of the target circuit may not be uniquely determined by its behavior [AACW06].

2.6 Queries

The learning algorithm gets information about the target circuit by specifying a value injection experiment e and observing the element of Σ assigned to the output wire. Such an action is termed a **value injection query**, abbreviated VIQ. A value injection query does not return complete information about the value distribution $C(e)$, but instead returns an element of Σ selected according to the distribution $C(e)$. Thus, in order to approximate the distribution $C(e)$, the learner must repeatedly make value injection queries with experiment e . In this case, the goal of learning is approximate behavioral equivalence.

2.7 The Learning Problem

The learning problem is ε -**approximate learning**: by making value injection queries to a target circuit C drawn from a known class of probabilistic circuits, find a circuit C' that is ε -behaviorally equivalent to C . The inputs to the learning algorithm are the names of the wires in C , the name of the output wire and positive numbers ε and δ , where the learning algorithm is required to succeed with probability at least $(1 - \delta)$.

3 Preliminary Results

In this section we establish some basic results about probabilistic circuits. We first note that if C is a probabilistic cir-

cuit, e is a distribution injection experiment and either $e(w)$ is a value distribution or e deterministically fixes all the input wires of w , then there is a value distribution D such that the value of w in $C(e)$ is determined by a random choice according to D , independent of the values chosen for any other wires. We make systematic use of this observation to reduce the number of experiments under consideration.

Lemma 1 *Let C_1 and C_2 be probabilistic circuits on wires W with the same output wire, let $w \in W$ be a wire, let D be a value distribution, and let e_1 and e_2 be distribution injection experiments such that $e_1(w) = e_2(w) = D$. Then there exists a value $\sigma \in \text{support}(D)$ such that*

$$d(C_1(e_1|_{w=\sigma}), C_2(e_2|_{w=\sigma})) \geq d(C_1(e_1), C_2(e_2)).$$

Proof: We have

$$\begin{aligned} & d(C_1(e_1), C_2(e_2)) \\ &= \frac{1}{2} \sum_{\tau \in \Sigma} \left| C_1(e_1)(\tau) - C_2(e_2)(\tau) \right| \\ &= \frac{1}{2} \sum_{\tau \in \Sigma} \left| \sum_{\rho \in \Sigma} C_1(e_1|_{w=\rho})(\tau) D(\rho) \right. \\ &\quad \left. - \sum_{\rho \in \Sigma} C_2(e_2|_{w=\rho})(\tau) D(\rho) \right| \\ &\leq \frac{1}{2} \sum_{\rho \in \Sigma} D(\rho) \sum_{\tau \in \Sigma} \left| C_1(e_1|_{w=\rho})(\tau) \right. \\ &\quad \left. - C_2(e_2|_{w=\rho})(\tau) \right| \\ &= \sum_{\rho \in \Sigma} D(\rho) d(C(e_1|_{w=\rho}), C(e_2|_{w=\rho})) \end{aligned}$$

by the triangle inequality. Let

$$\sigma = \arg \max_{\rho \in \text{support}(D)} d(C(e_1|_{w=\rho}), C(e_2|_{w=\rho})),$$

so that

$$d(C(e_1|_{w=\sigma}), C(e_2|_{w=\sigma})) \geq d(C(e_1), C(e_2))$$

by an averaging argument. ■

Lemma 2 *Let C_1 and C_2 be probabilistic circuits on wires W with the same output wire and let e be a distribution injection experiment. Then there exists a value injection experiment $e' \leq e$ such that*

$$d(C_1(e'), C_2(e')) \geq d(C_1(e), C_2(e)).$$

Proof: By induction on $|V|$, where $V \subseteq W$ is the set of wires that e constrains to nonconstant distributions. If $|V| > 0$, then let $w \in V$. By Lemma 1, there exists a value $\sigma \in \Sigma$ such that

$$d(C_1(e|_{w=\sigma}), C_2(e|_{w=\sigma})) \geq d(C_1(e), C_2(e)).$$

Since $e|_{w=\sigma}$ constrains one fewer wire to a nonconstant distribution, the existence of e' follows from the inductive hypothesis. ■

Corollary 3 *If circuits C_1 and C_2 are ε -behaviorally equivalent with respect to value injection experiments, then C_1 and C_2 are ε -behaviorally equivalent with respect to distribution injection experiments.*

Suppose that C is a probabilistic circuit and e_1 and e_2 are distribution injection experiments. For each wire w , we say that e_1 and e_2 **agree** on w if either

- e_1 and e_2 constrain w to the same distribution, or
- w is free in e_1 and e_2 , and e_1 and e_2 agree on all of w 's inputs.

If e_1 and e_2 agree on a wire w , then the marginal distributions of w in e_1 and e_2 are identical, that is, $C(e_1, w) = C(e_2, w)$.

Lemma 4 *Let C be a probabilistic circuit on wires W and let e_1 and e_2 be distribution injection experiments that agree on wires $V \subseteq W$. Then there exist distribution injection experiments $e'_1 \leq e_1$ and $e'_2 \leq e_2$ such that for each wire $w \in V$, there exists a value $\sigma \in \Sigma$ such that $e'_1(w) = e'_2(w) = \sigma$, and*

$$d(C(e'_1), C(e'_2)) \geq d(C(e_1), C(e_2)).$$

Proof: By induction on the number of unfixed wires $w \in V$. If there is such a wire, choose v to be one that is not reachable from the others. If $e_1(v) = e_2(v) = *$, then e_1 and e_2 agree on all of v 's inputs, and by the choice of v , all of v 's inputs are fixed. As such, we may assume without loss of generality that e_1 and e_2 in fact constrain v to the distribution $D = C(e_1, v) = C(e_2, v)$. By Lemma 1, there exists a value $\sigma \in \text{support}(D)$ such that

$$d(C(e_1|_{v=\sigma}), C(e_2|_{v=\sigma})) \geq d(C(e_1), C(e_2)).$$

The existence of e'_1 and e'_2 follows from the inductive hypothesis. ■

Lemma 5 *Let C be a probabilistic circuit on wires W , let e be a distribution injection experiment, let $w \in W$ be a wire free in e , and let D be a value distribution. Then e and $e|_{w=D}$ agree on all wires $u \in W$ to which there is no path on free wires from w .*

Proof: If u is constrained, then the conclusion follows. Otherwise, since u is free and has no free path from w , none of u 's inputs have free paths from w . We proceed by induction on the length of the longest path to u . If this length is zero, then u does not have any inputs. Otherwise, the inductive hypothesis applies to all of u 's inputs, on which e and $e|_{w=D}$ then must agree. It follows that they also agree on u . ■

Lemma 6 *Let C be a probabilistic circuit on wires W , let $w \in W$ be a wire, and let D_1, D_2 be value distributions. There exist value distributions D'_1, D'_2 with $\text{support}(D'_1) \cap \text{support}(D'_2) = \emptyset$ such that for all experiments e ,*

$$\begin{aligned} & d(C(e|_{w=D_1}), C(e|_{w=D_2})) \\ &= d(D_1, D_2) d(C(e|_{w=D'_1}), C(e|_{w=D'_2})). \end{aligned}$$

Proof: We have

$$\begin{aligned} d(C(e|_{w=D_1}), C(e|_{w=D_2})) &= \frac{1}{2} \sum_{\sigma \in \Sigma} \left| C(e|_{w=D_1})(\sigma) - C(e|_{w=D_2})(\sigma) \right| \\ &= \frac{1}{2} \sum_{\sigma \in \Sigma} \left| \sum_{\tau \in \Sigma} C(e|_{w=\tau})(\sigma) (D_1(\tau) - D_2(\tau)) \right|. \end{aligned}$$

If we let

$$\begin{aligned} \widehat{D}_1(\tau) &= D_1(\tau) - \min(D_1(\tau), D_2(\tau)) \\ \widehat{D}_2(\tau) &= D_2(\tau) - \min(D_1(\tau), D_2(\tau)), \end{aligned}$$

then

$$\begin{aligned} d(C(e|_{w=D_1}), C(e|_{w=D_2})) &= \frac{1}{2} \sum_{\sigma \in \Sigma} \left| \sum_{\tau \in \Sigma} C(e|_{w=\tau})(\sigma) (\widehat{D}_1(\tau) - \widehat{D}_2(\tau)) \right|. \end{aligned}$$

Since $\sum_{\tau \in \Sigma} \widehat{D}_1(\tau) = 1 - \sum_{\tau \in \Sigma} \min(D_1(\tau), D_2(\tau))$ and likewise for D_2 ,

$$\begin{aligned} d(D_1, D_2) &= \frac{1}{2} \sum_{\tau \in \Sigma} \left| D_1(\tau) - D_2(\tau) \right| \\ &= \frac{1}{2} \sum_{\tau \in \Sigma} \left| \widehat{D}_1(\tau) - \widehat{D}_2(\tau) \right| \\ &= \sum_{\tau \in \Sigma} \widehat{D}_1(\tau) = \sum_{\tau \in \Sigma} \widehat{D}_2(\tau). \end{aligned}$$

If $d(D_1, D_2) > 0$, then the distributions D'_1 and D'_2 where

$$\begin{aligned} D'_1(\tau) &= \widehat{D}_1(\tau) / d(D_1, D_2) \\ D'_2(\tau) &= \widehat{D}_2(\tau) / d(D_1, D_2) \end{aligned}$$

satisfy the requisite properties. Otherwise, any two distributions with disjoint support will do. ■

4 Test Paths

The concept of a test path has been central in previous work on learning deterministic circuits by means of value injection queries [AACR07, AACW06]. A **test path** for a wire w is a value injection experiment in which the free gates form a directed path in the circuit graph from w to the output wire. All the other wires in the circuit are fixed; this includes the inputs of w . A **side wire** with respect to a test path p is a wire fixed by p that is input to a free wire in p . A test path may help the learning algorithm determine the effects of assigning different values to the wire w . The test-path lemmas from [AACR07, AACW06] may be re-stated as follows.

Lemma 7 *Let C be a deterministic circuit. If for some value injection experiment e , wire w and alphabet symbols σ and τ it is the case that*

$$C(p|_{w=\sigma}) = C(p|_{w=\tau})$$

for every test path $p \leq e$ then also

$$C(e|_{w=\sigma}) = C(e|_{w=\tau}).$$

Nontrivial complications arise in attempting to carry over this test path lemma to general probabilistic circuits, as we now show. The following lemma shows that for alphabets of size at least four, there are transitively reduced probabilistic circuits for which the test-path lemma fails completely. (A less intuitive version of this construction shows that this phenomenon occurs also at alphabet size three.)

Lemma 8 *If $|\Sigma| = 4$, there exists a probabilistic circuit C , value injection experiment e , wire w and alphabet symbols σ and τ such that although for every test path $p \leq e$ for w , $d(C(p|_{w=\sigma}), C(p|_{w=\tau})) = 0$, it is nevertheless the case that $d(C(e|_{w=\sigma}), C(e|_{w=\tau})) = 1$.*

Proof: Assume $\Sigma = \{00, 01, 10, 11\}$, and define probabilistic gate functions T, L, R , and X as follows.

$$\begin{aligned} T(00) = T(11) &= U(\{00, 11\}), \\ T(01) = T(10) &= U(\{01, 10\}), \\ L(00) = L(01) &= 00, \\ L(10) = L(11) &= 01, \\ R(00) = R(10) &= 00, \\ R(01) = R(11) &= 01, \end{aligned}$$

and $X(ab, cd) = 0(b \oplus d)$, where \oplus is sum modulo 2.

The circuit C has 5 wires, connected as in Figure 2. The output wire is w_5 ; note that C is transitively reduced.

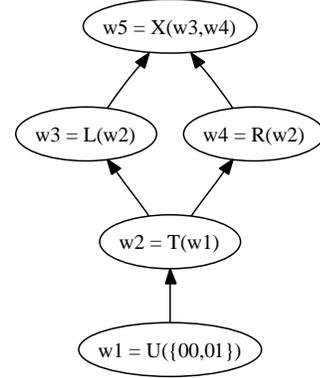


Figure 2: The circuit C ; w_5 is the output wire.

Consider the experiment e that leaves all the wires free. We have $C(e|_{w_1=00}) = 00$ and $C(e|_{w_1=01}) = 01$, and thus $d(C(e|_{w_1=00}), C(e|_{w_1=01})) = 1$. However, the only test paths for w_1 fix w_3 and leave all other wires free or fix w_4 and leave all other wires free. Calculation verifies that fixing w_3 or w_4 to any value and leaving the other wires free yields the output distribution $U(\{00, 01\})$ regardless of whether w_1 is fixed to 00 or 01. Thus, for every test path p for w_1 , we have $d(C(p|_{w_1=00}), C(p|_{w_1=01})) = 0$. ■

4.1 A Bound for Boolean Probabilistic Circuits

Surprisingly, for Boolean probabilistic circuits there is a useful quantitative relationship between the differences exposed by test paths and the differences exposed by arbitrary experiments.

Let e be an experiment and w a wire. Define $\Pi(e, w)$ to be the set of all directed paths from w to the output wire on free wires in e . Let $S(e)$ be the set of wires that originate a free shortcut, that is, the set of free wires w such that there exists a path $p \in \Pi(e, w)$ with two free wires to which w is an input. Define

$$\kappa(e, w) = \sum_{p \in \Pi(e, w)} 2^{|p \cap S(e)|}.$$

Lemma 9 *Let C be a probabilistic circuit, e be a distribution injection experiment, w and u be free wires where w is an input to u , and D_0 be a value distribution. Let $\beta = 2$ if $w \in S(e)$ and $\beta = 1$ otherwise. Then*

$$\kappa(e, w) = \kappa(e|_{u=D_0}, w) + \kappa(e|_{w=1}, u) \cdot \beta.$$

Proof: The first term of the sum counts paths that don't contain u , and the second counts paths that do. Let $e' = e|_{u=D_0}$ and $e'' = e|_{w=1}$. We have

$$\begin{aligned} \kappa(e, w) &= \sum_{p \in \Pi(e, w)} 2^{|p \cap S(e)|} \\ &= \sum_{\substack{p \in \Pi(e, w) \\ u \notin p}} 2^{|p \cap S(e)|} + \sum_{\substack{p \in \Pi(e, w) \\ u \in p}} 2^{|p \cap S(e)|} \\ &= \sum_{p \in \Pi(e', w)} 2^{|p \cap S(e')|} + \sum_{p \in \Pi(e'', u)} 2^{|p \cap S(e'')|} \beta \\ &= \kappa(e', w) + \kappa(e'', u) \cdot \beta, \end{aligned}$$

since each path $p \ni u$ from w corresponds to the path $p \setminus \{w\}$ from u . \blacksquare

Lemma 10 *Let C be a Boolean probabilistic circuit, e be a distribution injection experiment, w be a wire, and D_1, D_2 be value distributions. If there exists $\varepsilon \geq 0$ such that for all w -test paths $p \leq e$,*

$$d(C(p|_{w=D_1}), C(p|_{w=D_2})) \leq \varepsilon,$$

then

$$d(C(e|_{w=D_1}), C(e|_{w=D_2})) \leq \kappa(e, w) \cdot \varepsilon.$$

Proof: By induction on $\phi(e)$, the number of free wires in e . By Lemma 6, assume that $\text{support}(D_1) \cap \text{support}(D_2) = \emptyset$. The critical feature of the Boolean case is that it follows that $D_1 = 0$ and $D_2 = 1$ without loss of generality—it is important to the following proof that D_1 and D_2 be deterministic.

If $\phi(e) = 1$, then either

$$d(C(e|_{w=0}), C(e|_{w=1})) = 0,$$

or w is the output, e is a w -test path, and $\kappa(e, w) = 1$. Otherwise, the inductive hypothesis is that the lemma holds for all experiments e' with $\phi(e') < \phi(e)$.

Except for w , the experiments $e|_{w=0}$ and $e|_{w=1}$ agree on all constrained wires, so by Lemmas 4 and 5, assume without loss of generality that every wire with no free path from w is in fact fixed. Since C is acyclic, there exists a free wire $u \neq w$ whose only unfixed input is w . Let g be the gate assigned

by C to u and let $B_0 = g(e|_{w=0})$ and $B_1 = g(e|_{w=1})$, so that

$$\begin{aligned} C(e|_{w=0}) &= C(e|_{w=0, u=B_0}) \\ C(e|_{w=1}) &= C(e|_{w=1, u=B_1}). \end{aligned}$$

By the triangle inequality,

$$\begin{aligned} d(C(e|_{w=0}), C(e|_{w=1})) &\leq d(C(e|_{w=0, u=B_0}), C(e|_{w=1, u=B_0})) \\ &\quad + d(C(e|_{w=1, u=B_0}), C(e|_{w=1, u=B_1})). \end{aligned}$$

The inductive hypothesis bounds the first term of the sum by $\kappa(e', w) \cdot \varepsilon$, where $e' = e|_{u=B_0}$. We now derive a bound on u -test paths so that the inductive hypothesis applies to the second term as well. Let $\beta = 2$ if $w \in S(e)$ and $\beta = 1$ otherwise. Let $e'' = e|_{w=1}$ and suppose $p \leq e''$ is a u -test path. Then

$$\begin{aligned} d(C(p|_{u=B_0}), C(p|_{u=B_1})) &\leq d(C(p|_{w=1, u=B_0}), C(p|_{w=0, u=B_0})) \\ &\quad + d(C(p|_{w=0, u=B_0}), C(p|_{w=1, u=B_1})) \\ &= d(C(p|_{w=0, u=B_0}), C(p|_{w=1, u=B_0})) \\ &\quad + d(C(p|_{w=0, u=*}), C(p|_{w=1, u=*})) \\ &\leq \beta \varepsilon, \end{aligned}$$

since both terms of the sum are bounded by ε , and the first is nonzero only if w is an input to some free wire in p other than u . Thus

$$d(C(e''|_{u=0}), C(e''|_{u=1})) \leq \kappa(e'', u) \cdot \beta \varepsilon,$$

and,

$$\begin{aligned} d(C(e|_{w=0}), C(e|_{w=1})) &\leq \kappa(e', w) \cdot \varepsilon + \kappa(e'', u) \cdot \beta \varepsilon \\ &= \kappa(e, w) \cdot \varepsilon, \end{aligned}$$

by Lemma 9. \blacksquare

In the case of transitively reduced circuits, $S(e) = \emptyset$, and $\kappa(e, w) = \pi(e, w)$, where $\pi(e, w) = |\Pi(e, w)|$, the number of directed paths on free wires in e from w to the output wire.

Corollary 11 *Let C be a transitively reduced Boolean probabilistic circuit, e be a distribution injection experiment, and w be a wire. If there exists $\varepsilon \geq 0$ such that for all w -test paths $p \leq e$,*

$$d(C(p|_{w=0}), C(p|_{w=1})) \leq \varepsilon,$$

then

$$d(C(e|_{w=0}), C(e|_{w=1})) \leq \pi(e, w) \cdot \varepsilon.$$

5 Learning Boolean Probabilistic Circuits

The amount of attenuation given by Lemma 10 allows us to adapt the CircuitBuilder algorithm [AACW06] to learn Boolean probabilistic circuits with constant fan-in and log depth in polynomial time.

Theorem 12 *Given constants c and k there is a nonadaptive learning algorithm that with probability at least $(1 - \delta)$ successfully ε -approximately learns any Boolean probabilistic circuit with n wires, gates of fan-in at most k and depth at most $c \log n$ using value injection queries in time bounded by a polynomial in n , $1/\varepsilon$ and $\log(1/\delta)$.*

We adapt the Circuit Builder algorithm from [AACW06] to prove Theorem 12 and call the resulting algorithm Probabilistic Circuit Builder (PCB). The algorithm constructs a set U of experiments such that every test path is equivalent to some experiment in U , obtains a sufficiently good estimate of the output distribution for each experiment in U , and then builds a circuit approximately behaviorally equivalent to the target circuit by repeatedly adding sufficiently accurate gates all of whose inputs are in the partially constructed circuit.

Let the target circuit be C and let positive constants δ , ε , k and c be given such that the fan-in of C is bounded by k and the depth of C is bounded by $c \log n$. For such a circuit, $\pi(e, w)$ is bounded above by $k^{c \log n}$, so the quantity $\kappa(e, w)$ is bounded above by

$$\kappa(n) = k^{c \log n} \cdot 2^{c \log n} = n^{c(\log k + 1)} = n^{O(1)}.$$

The PCB algorithm is nonadaptive: it computes a set U of value injection experiments, repeats each value injection query for $e \in U$ sufficiently many times to estimate the expected value of $C(e)$ with enough accuracy, and then uses the results of the queries to build a circuit C' that is ε -behaviorally equivalent to C .

In choosing the experiments U , the goal is that for every potential test path, U includes an equivalent experiment. The structure of the circuit, however, is not known *a priori*, a difficulty that we overcome by the same method as [AACW06]. Let U_* be a universal set of value injection experiments such that for every set of $kc \log n$ wires and every assignment of symbols from $\Sigma \cup \{*\}$ to those wires, some experiment $e \in U_*$ agrees with the values assigned to those wires. As in [AACW06], it is possible to construct such a set U of size

$$2^{O(kc \log n)} \log n = n^{O(kc)}$$

in time polynomial in its size.

For every wire w and test path p for w , there is an experiment in U_* that leaves the path wires of p free and fixes the side wires of p to their values in p . Consequently, p and this experiment agree on the output wire. Although it is tempting now to set $U = U_*$, there is no easy way to determine which experiment a test path corresponds to, making it difficult for PCB to perform comparisons where w is fixed to different values. For $b = 0, 1$, then, let U_b contain every experiment $e|_{w=b}$ such that $e \in U_*$ and w is free in e . Now we can take $U = U_* \cup U_0 \cup U_1$.

For each $e \in U$, PCB repeatedly makes a value injection query with e to estimate the distribution of $C(e)$. By Hoeffding's bound, we have that

$$m = O((n\kappa(n)/\varepsilon)^2 \log(|U|/\delta))$$

trials per experiment e suffice to guarantee that with probability at least $1 - \delta$, for all $e \in U$,

$$d(C(e), \widehat{C}(e)) \leq \varepsilon/(5n\kappa(n)). \quad (1)$$

If (1) holds, then we can compute good estimates for a class of distribution experiments. Let $e \in U_*$ be a value injection experiment, w be a wire that e leaves free, and D be a value distribution. Then let

$$\widehat{C}(e|_{w=D}) = \sum_{\sigma \in \Sigma} D(\sigma) \widehat{C}(e|_{w=\sigma}).$$

We have

$$\begin{aligned} & d(C(e|_{w=D}), \widehat{C}(e|_{w=D})) \\ & \leq \sum_{\sigma \in \Sigma} D(\sigma) d(C(e|_{w=\sigma}), \widehat{C}(e|_{w=\sigma})) \\ & \leq \varepsilon/(5n\kappa(n)). \end{aligned}$$

From this point on, we assume that the estimates are correct and show that PCB successfully builds a circuit C' that is ε -behaviorally equivalent to C .

PCB builds the circuit C' one gate at a time. Initially C' has no gates assigned to wires. The algorithm tries repeatedly to find a wire w and a gate g such that g is ε/n -correct for w in C and all of g 's inputs are in C' . When this is no longer possible, PCB outputs C' and halts.

To prove the correctness of PCB, we first establish two lemmas connecting gates, paths and experiments. Given a Boolean probabilistic circuit C and a probabilistic gate g , g is η -**correct** for wire w with respect to C if for every value injection experiment e that fixes the input wires for g we have $d(C(e), C(e|_{w=g(e)})) \leq \eta$, where $g(e)$ denotes the coin flip determined by g when its inputs are fixed as in e . Recall that $\phi(e)$ denotes the number of free wires in experiment e .

Lemma 13 *Let C and C' be probabilistic circuits on wires W , and let e be a distribution injection experiment. If for every wire w , the gate g for w in C' is η -correct for w with respect to C , then*

$$d(C(e), C'(e)) \leq \phi(e) \cdot \eta.$$

Proof: By induction on $\phi(e)$, the number of free wires in e . If $\phi(e) = 0$, then e constrains the output wire, and trivially, $d(C(e), C'(e)) = 0$. Otherwise, the inductive hypothesis is that C and C' are η -behaviorally equivalent with respect to all experiments with fewer free gates.

By Lemma 2, assume that e is in fact a value injection experiment. Since C' is acyclic, there exists a free wire w in e such that the inputs to w in C' are fixed in e to some k -tuple $(\sigma_1, \dots, \sigma_k) \in \Sigma^k$. Letting f be the probabilistic gate function for w in C' , we have $C'(e) = C'(e|_{w=f(\sigma_1, \dots, \sigma_k)})$, and

$$\begin{aligned} & d(C(e), C'(e)) \\ & \leq d(C(e), C(e|_{w=f(\sigma_1, \dots, \sigma_k)})) \\ & \quad + d(C(e|_{w=f(\sigma_1, \dots, \sigma_k)}), C'(e|_{w=f(\sigma_1, \dots, \sigma_k)})) \\ & \leq \eta + (\phi(e) - 1) \cdot \eta = \phi(e) \cdot \eta \end{aligned}$$

by the fact that f is η -correct and the inductive hypothesis. ■

Next we show that test paths are sufficient to determine whether a gate is η -correct for a wire in C .

Lemma 14 *Let C be a Boolean probabilistic circuit, w a wire and g' a probabilistic gate. If for every test path p for w that fixes all the inputs of g' , $d(C(p), C(p|_{w=g'(p)})) \leq \eta/\kappa(C)$, where $\kappa(C)$ is the maximum value of $\kappa_{C'}(e, w)$ over all circuits C' with the same set of wires, all experiments e , and all wires w , then g' is η -correct for w with respect to C .*

Proof: Let g be the actual gate that C assigns to w . Let e be a value injection experiment that fixes every input of g' . e may not fix all of g 's inputs, but since C is acyclic, g 's inputs are not reachable from w . By Lemmas 4 and 5, there exists an experiment $e' \leq e$ that fixes g 's inputs, with

$$d(C(e'), C(e'|_{w=g'(e')})) \geq d(C(e), C(e|_{w=g'(e)})).$$

Since e' fixes all of g 's inputs, $C(e') = C(e'|_{w=g'(e')})$. It is given that for all test paths p that fix all inputs of g and g' that

$$d(C(p|_{w=g(p)}), C(p|_{w=g'(p)})) \leq \eta/\kappa(C),$$

so it follows by Lemma 10 that

$$\begin{aligned} d(C(e'|_{w=g'(e')}), C(e'|_{w=g'(e')})) & \\ & \leq \kappa(e', w) \cdot \eta/\kappa(C) \\ & \leq \eta, \end{aligned}$$

and g' is η -correct for w . \blacksquare

To prove the correctness of PCB, we argue as follows. Let V be the set of wires to which C' does not assign a gate. Then since C is acyclic, there is some wire $w \in V$ such that none of w 's inputs in C belong to V . PCB looks for a gate g' such that for each experiment $e \in U_*$ that leaves w free and fixes all inputs of g' ,

$$d(\widehat{C}(e), \widehat{C}(e|_{w=g'(e)})) \leq 3\varepsilon/(5n\kappa(n)). \quad (2)$$

Then

$$d(C(e), \widehat{C}(e)) \leq \varepsilon/(5n\kappa(n))$$

$$d(\widehat{C}(e|_{w=g'(e)}), C(e|_{w=g'(e)})) \leq \varepsilon/(5n\kappa(n)),$$

and

$$d(C(e|_{w=g'(e)}), C(e|_{w=g(e)})) \leq \varepsilon/(n\kappa(n))$$

by (1) and the triangle inequality. It follows by Lemma 14 that g' is ε/n -correct for w in C . Let g be the gate that C assigns to w and suppose that $d(g(e), g'(e)) \leq \varepsilon/(5n\kappa(n))$ for all experiments e that fix g 's inputs. Then

$$d(\widehat{C}(e), C(e)) \leq \varepsilon/(5n\kappa(n))$$

$$d(C(e), C(e|_{w=g(e)})) = 0$$

$$d(C(e|_{w=g(e)}), C(e|_{w=g'(e)})) \leq \varepsilon/(5n\kappa(n))$$

$$d(C(e|_{w=g'(e)}), \widehat{C}(e|_{w=g'(e)})) \leq \varepsilon/(5n\kappa(n))$$

and g' satisfies (2). Therefore, PCB will continue to make progress.

To bound the running time of PCB we argue as follows. The set U of experiments is of cardinality $n^{O(kc)}$ and can be constructed in time polynomial in its size. Each experiment in U is repeated

$$O((n\kappa(n)/\varepsilon)^2 \log(|U|/\delta))$$

times; recall that $\kappa(n) = O(n^{c(\log k+1)})$. PCB chooses a gate for a wire n times. Each gate it tests must be subjected to a polynomial number of experiments; in order to be assured of a sufficiently good approximation, it must iterate over $O(n^k)$ sets of inputs times $|\Sigma|^k$ entries times a polynomial number of points in $[0, 1]^\Sigma$ to be assured of finding a sufficiently good approximation to a true gate. Thus the running time of PCB is polynomial in n , $1/\varepsilon$ and $1/\delta$.

6 Lower Bounds

We consider lower bounds on the path attenuation factors for Boolean probabilistic circuits. The following lemma shows that the bound of $\pi(e, w)$ for transitively reduced Boolean probabilistic circuits in Corollary 11 is tight infinitely often.

Lemma 15 *There is an infinite set of transitively reduced probabilistic Boolean circuits such that for each circuit C in the family, there exists a value injection experiment e and a wire w such that*

$$d(C(e|_{w=0}), C(e|_{w=1})) = 1$$

and for every test path p for w we have

$$d(C(p|_{w=0}), C(p|_{w=1})) = 1/\pi(e, w).$$

Proof: For each positive integer ℓ , define the circuit C_ℓ to be a chain of ℓ copies of the circuit C_1 in Figure 1 with wire w_4 of one copy identified with wire w_1 of the next copy. More formally, the $3d + 1$ wires are $w_{0,4}$ and $w_{i,j}$ for $i = 1, \dots, d$ and $j = 2, 3, 4$. The output wire is $w_{d,4}$. The wire $w_{0,4}$ has no inputs and is determined by an unbiased coin flip, that is, $U(\{0, 1\})$. The wires $w_{i,2}$ and $w_{i,3}$ are the outputs of deterministic identity gates with input $w_{i-1,4}$. The wire $w_{i,4} = A(w_{i,2}, w_{i,3})$ is the result of applying the two-input averaging gate A to the wires $w_{i,2}$ and $w_{i,3}$.

The experiment e leaves all of the wires free. Let w denote the wire $w_{0,4}$. Clearly there are 2^ℓ paths on free gates in e from w to the output gate, that is, $\pi(w, e) = 2^\ell$. For experiment e we have $C(e|_{w=0}) = 0$ and $C(e|_{w=1}) = 1$, so $d(C(e|_{w=0}), C(e|_{w=1})) = 1$. However, any test path p for w must fix one of the wires $w_{i,2}$ or $w_{i,3}$ for each $i = 1, \dots, d$. As the signal proceeds through each level, it is attenuated by $1/2$, so the final result for any test path p for w is $d(C(p|_{w=0}), C(p|_{w=1})) = 1/2^\ell = 1/\pi(e, w)$. \blacksquare

A generalization of this construction shows that for any transitively reduced circuit graph, there is an assignment of Boolean probabilistic functions that matches the attenuation factor of $\pi(e, w)$.

Lemma 16 *Let G be a transitively reduced directed graph with a designated output node in which there is a path from every node to the output node. There is a Boolean probabilistic circuit C whose circuit graph is G such that for every value injection experiment e and for every test path $p \leq e$ and every wire w ,*

$$\begin{aligned} d(C(e|_{w=1}), C(e|_{w=0})) \\ & \geq \pi(e, w) \cdot d(C(p|_{w=1}), C(p|_{w=0})). \end{aligned}$$

Proof: (Proof omitted in this abstract.) \blacksquare

Can the general bound in Lemma 10 be improved to the bound for transitively reduced circuits in Corollary 11? The following example shows that the better bound is in general not attainable if the circuit is not transitively reduced. It gives a family of circuits of depth $2d$ for which the worst-case ratio of the differences shown for w by an experiment e and the best path for w is $(5/4)^d \pi(e, w)$.

Lemma 17 *There exists an infinite set of Boolean probabilistic circuits D_1, D_2, \dots such that for each ℓ there exists a value injection experiment e and a wire w such that $\pi(e, w) = 4^\ell$ and*

$$d(D_\ell(e|_{w=0}), D_\ell(e|_{w=1})) = (5/7)^\ell,$$

but for any test path p for w ,

$$d(D_\ell(p|_{w=0}), D_\ell(p|_{w=1})) = (1/7)^\ell.$$

Proof: We first define a Boolean probabilistic circuit D_1 and then connect ℓ copies of it in series to get D_ℓ . The wires of D_1 are w_1, \dots, w_5 . They are connected as in Figure 3; the output wire is w_5 . Note that the edge (w_1, w_5) means that the circuit graph is not transitively reduced. The gate function G

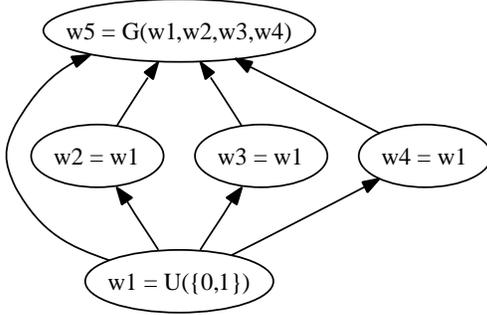


Figure 3: The circuit D_1 ; w_5 is the output wire.

is defined by giving its expected value as a function of its inputs:

$$E[G(w_1, w_2, w_3, w_4)] = ((1 - w_1) + 2w_2 + 2w_3 + 2w_4)/7.$$

Let e be the experiment that leaves all five wires free. It is clear that

$$d(D_1(e|_{w=0}), D_1(e|_{w=1})) = 5/7.$$

We now show that for any test path p for w_1 ,

$$d(D_1(p|_{w=0}), D_1(p|_{w=1})) = 1/7.$$

The possible test paths p for w_1 either fix all of w_2, w_3, w_4 or all but one of them. Thus, as we change from $w_1 = 0$ to $w_1 = 1$ in such a test path, the assignments to wires (w_1, w_2, w_3, w_4) change in one of four possible ways:

$$(0, b_2, b_3, b_4) \text{ to } (1, b_2, b_3, b_4)$$

$$(0, 0, b_3, b_4) \text{ to } (1, 1, b_3, b_4)$$

$$(0, b_2, 0, b_4) \text{ to } (1, b_2, 1, b_4)$$

$$(0, b_2, b_3, 0) \text{ to } (1, b_2, b_3, 1)$$

Checking each of these possible changes against the definition of G , we see that each change produces a difference of $1/7$, as claimed. (This example can be modified to give a difference of 1 versus $1/5$; details are omitted in this abstract.) Thus, D_1 gives the base case of the claim in the lemma.

To construct D_ℓ , we take ℓ copies of D_1 and identify wire w_5 in one copy with wire w_1 in the next copy, making the wire w_5 of the final copy the output wire of the whole

circuit. Let w denote the wire w_1 in the first such copy. Then $\pi(e, w) = 4^\ell$ and

$$d(D_\ell(e|_{w=0}), D_\ell(e|_{w=1})) = (5/7)^\ell.$$

For any test path p , the signal is attenuated by a factor of $1/7$ for each level, and we have

$$d(D_\ell(p|_{w=0}), D_\ell(p|_{w=1})) = 1/7^\ell. \quad \blacksquare$$

The construction can be generalized to $k+1$ wires for any odd $k+1$, which increases the attenuation. In the base circuit there are k paths and an attenuation factor of $1/(2k-3)$, and the worst-case ratio of differences for an experiment and its test paths in D_ℓ approaches $2^\ell \pi(e, w)$ as k goes to infinity.

7 Non-Boolean Circuits Revisited

The sharp contrast in results for transitively reduced circuits with alphabet size at least three, for which test paths may show no difference (Lemma 8) and those with alphabet size two, for which test paths must show a significant difference (Lemma 10) motivate us to consider a generalization of the kinds of experiments we consider, to function injection experiments. This generalization allows us to extend the results of Lemma 10 to non-Boolean alphabets.

In a value injection experiment, each wire is either fixed to a constant value or left free. In a function injection experiment, these possibilities are expanded to permit a transformation of the value that the wire would take if it were left free. As an example, consider a transformation in which the values are linearly ordered and all values below a certain threshold are mapped to the minimum value and all other values are mapped to the maximum value. It is conceivable that this kind of transformation could be feasible in some domains; in any case, the theoretical consequences are quite interesting. We first give a general definition of function injection, but in the results below we are primarily concerned with 2-partitions, that is, transformations that are like the above example in that they partition the values into two blocks and map each block to a fixed element of the block.

An **alphabet transformation** is a function f that maps symbols to distributions over symbols. An alphabet transformation is **deterministic** if it assigns only deterministic distributions, in which case we think of it as a map from symbols to symbols. A deterministic alphabet transformation f is a **k -partition** if there exists a partition of Σ into at most k disjoint nonempty sets Σ_i such that for each i there exists $\sigma_i \in \Sigma_i$ such that $f(\Sigma_i) = \{\sigma_i\}$. We use 2-partitions to reduce the case of larger alphabets to the binary case. Note that the 2-partitions of a binary alphabet include the identity and the two constant functions, but not the negation function.

If D is a value distribution and f is an alphabet transformation, then $f(D)$ is the value distribution in which

$$(f(D))(\sigma) = \sum_{\tau \in \Sigma} D(\tau)(f(\tau))(\sigma).$$

A **function injection experiment** is a mapping e with domain W that assigns to each wire the symbol $*$ or a symbol from Σ or an alphabet transformation f . Then e leaves w **free** if $e(w) = *$, **fixes** w if $e(w) \in \Sigma$, and **transforms**

w if $e(w)$ is an alphabet transformation f . We extend the ordering \leq on experiments by stipulating that each alphabet transformation $f \leq *$. A **2-partition experiment** is a function injection experiment in which every alphabet transformation is a 2-partition.

We now define the joint probability distribution on assignments of symbols from Σ to wires determined by a function injection experiment e . If e fixes w , then w is just assigned $e(w)$. Otherwise, if the inputs of w have been assigned the values $\sigma_1, \dots, \sigma_k$ and f is the gate function for w , we randomly and independently choose a symbol σ according to the value distribution $f(\sigma_1, \dots, \sigma_k)$. If w is free in e , then σ is the symbol assigned to w ; however, if $e(w)$ is an alphabet transformation, then a symbol τ is chosen randomly and independently according to the value distribution $e(\sigma)$ and assigned to w . That is, when $e(w)$ is an alphabet transformation, we generate the symbol for w as though it were free, and then use the distribution $e(w)$ to transform that symbol. Because C is acyclic, this process assigns a symbol to every wire of C .

In a **function injection query** (FIQ), the learning algorithm gives a function injection experiment e and receives a symbol σ assigned to the output wire of C by the probability distribution defined above. A **functional test path** for a wire w is a function injection experiment in which the free and transformed wires are a directed path in the circuit graph from w to the output wire, and all other wires are fixed.

As an example of how functional test paths help in learning non-Boolean probabilistic circuits, consider the circuit in the proof of Lemma 8. We specify a functional test path p by $p(w_1) = p(w_3) = p(w_5) = *$, $p(w_4) = 00$ and $p(w_2)$ is the alphabet transformation $00 \rightarrow 00$, $01 \rightarrow 01$, $10 \rightarrow 01$, and $11 \rightarrow 00$. Note that the alphabet transformation is a 2-partition. Then $C(p|_{w_1=00}) = 00$ but $C(p|_{w_1=01}) = 01$ deterministically, so this functional test path witnesses a difference of 1, as large as the experiment that leaves all the wires free. Test paths with functions allow us to carry over the results of Lemma 10 to non-Boolean alphabets.

Lemma 18 *Let C be a probabilistic circuit, e be a function injection experiment, w be a wire, and D_1, D_2 be value distributions. If there exists $\varepsilon \geq 0$ such that for all functional w -test paths $p \leq e$,*

$$d(C(p|_{w=D_1}), C(p|_{w=D_2})) \leq \varepsilon,$$

then

$$d(C(e|_{w=D_1}), C(e|_{w=D_2})) \leq \kappa(e, w) \cdot \varepsilon.$$

Proof: The obstacle in Lemma 10 is that when the alphabet is non-Boolean, we may assume only that D_1 and D_2 have disjoint support, not that they are deterministic. This obstacle can be overcome by injecting a 2-partition at w . Let $\Sigma_1 = \text{support}(D_1)$ and $\Sigma_2 = \text{support}(D_2)$ and assume $\Sigma_1 \cap \Sigma_2 = \emptyset$. Then

$$\begin{aligned} & d(C(e|_{w=D_1}), C(e|_{w=D_2})) \\ & \leq \sum_{\substack{\rho_1 \in \Sigma_1 \\ \rho_2 \in \Sigma_2}} D_1(\rho_1) D_2(\rho_2) d(C(e|_{w=\rho_1}), C(e|_{w=\rho_2})) \end{aligned}$$

by the triangle inequality. Let

$$(\sigma, \tau) = \arg \max_{\substack{\rho_1 \in \Sigma_1 \\ \rho_2 \in \Sigma_2}} d(C(e|_{w=\rho_1}), C(e|_{w=\rho_2}))$$

so that

$$\begin{aligned} & d(C(e|_{w=D_1}), C(e|_{w=D_2})) \\ & \leq d(D_1, D_2) d(C(e|_{w=\sigma}), C(e|_{w=\tau})). \end{aligned}$$

Let f be an alphabet transformation that maps Σ_1 to σ and Σ_2 to τ and all other symbols to either σ or τ . Then f is a 2-partition, and

$$\begin{aligned} & d(C(e|_{w=D_1}), C(e|_{w=D_2})) \\ & \leq d(C(e|_{w=f(D_1)}), C(e|_{w=f(D_2)})). \end{aligned}$$

Since $f(D_1) = \sigma$ and $f(D_2) = \tau$, the rest of the proof goes through. ■

Corollary 19 *Let C be a transitively reduced probabilistic circuit, e be a function injection experiment, w be a wire, and D_1, D_2 be value distributions. If there exists $\varepsilon \geq 0$ such that for all functional w -test paths $p \leq e$,*

$$d(C(p|_{w=D_1}), C(p|_{w=D_2})) \leq \varepsilon,$$

then

$$d(C(e|_{w=D_1}), C(e|_{w=D_2})) \leq \pi(e, w) \cdot \varepsilon.$$

Certain natural questions arise in response to the introduction of function injection experiments. We can define circuits C and C' to be **strongly behaviorally equivalent** if $C(e) = C'(e)$ for every function injection query e . Does behavioral equivalence imply strong behavioral equivalence? Once again, alphabet size determines the answer: no for alphabet size greater than two, yes for alphabet size two.

Lemma 20 *For $\Sigma = \{0, 1, 2\}$, there exist deterministic circuits C_1 and C_2 that are behaviorally equivalent but not strongly behaviorally equivalent.*

Proof: In both C_1 and C_2 there are two wires w_1 and w_2 , where w_2 is the output wire. In both circuits the gate for w_2 has input w_1 and deterministically maps 0 to 0 and maps 1 and 2 to 1. In C_1 , w_1 is the constant 1 and C_2 it is the constant 2.

Then if e is the value injection experiment that leaves both wires free, $C_1(e) = 1 = C_2(e)$. If e fixes either w_1 or w_2 , then also $C_1(e) = C_2(e)$. Thus C_1 is behaviorally equivalent to C_2 .

However, the 2-partition function injection experiment e that leaves w_2 free and maps the output of w_1 according to the transformation $0 \rightarrow 0$, $1 \rightarrow 0$, $2 \rightarrow 2$ yields $C_1(e) = 0$ and $C_2(e) = 1$. Thus C_1 is not strongly behaviorally equivalent to C_2 . ■

However, 2-partition function experiments suffice to establish strong behavioral equivalence.

Lemma 21 *Let C and C' be probabilistic circuits with the same alphabet Σ , the same set of wires and the same output wire. If $C(e) = C'(e)$ for every 2-partition function experiment e then C and C' are strongly behaviorally equivalent.*

Proof: By another modification of the proof of Lemma 10. ■

Because in the Boolean case every 2-partition function injection query is a value injection query, we have the following.

Corollary 22 *For Boolean probabilistic circuits C and C' , if C is behaviorally equivalent to C then C' is strongly behaviorally equivalent to C' .*

8 Discussion and Open Problems

These results concern general probabilistic acyclic circuits, with no restriction other than fan-in on the kinds of probabilistic gates considered. Particular domains may warrant specific assumptions about the gates, which may make the learning problems more tractable. For example, for the problem of learning the structure of an independent cascade social network using exact value injection queries, a query-optimal algorithm is presented in [AAR]. Note that the networks in this domain may contain cycles, which complicates their analysis.

Initial work suggests that Corollary 11 allows us to adapt the Distinguishing Paths algorithm [AACR07] to learn transitively reduced Boolean probabilistic circuits, given a bound on the number of paths in the circuit graph. We would like to adapt Circuit Builder to use functional test paths to learn non-Boolean circuits; in this case the universal set must map wires to the set containing all alphabet symbols from Σ and all 2-partitions of Σ , of which there are fewer than $|\Sigma|^{2^2|\Sigma|}$. Thus, the universal set will still be of size $n^{O(1)}$, suggesting that a polynomial time algorithm may be attainable in this case. An open question is whether not-injection reduces the maximum path attenuation to just the number of paths for general Boolean probabilistic circuits. A very interesting direction of future work is whether there are computationally feasible approaches to learning probabilistic circuits that use experiments more general than paths and thereby avoid the problem of path attenuation.

9 Acknowledgments

This work was done while Jiang Chen was a member of the Center for Computational Learning Systems, Columbia University. The authors thank the reviewers of the present paper for their thoughtful comments.

References

- [AACR07] Dana Angluin, James Aspnes, Jiang Chen, and Lev Reyzin. Learning large-alphabet and analog circuits with value injection queries. In *the 20th Annual Conference on Learning Theory*, pages 51–65, 2007.
- [AACW06] Dana Angluin, James Aspnes, Jiang Chen, and Yinghua Wu. Learning a circuit by injecting values. In *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*, pages 584–593, New York, NY, USA, 2006. ACM Press.
- [AAR] Dana Angluin, James Aspnes, and Lev Reyzin. Optimally learning social networks with activations and suppressions. Submitted to COLT 2008.
- [AK95] Dana Angluin and Michael Kharitonov. When won't membership queries help? *J. Comput. Syst. Sci.*, 50(2):336–355, 1995.
- [AKMM98] Tatsuya Akutsu, Satoru Kuhara, Osamu Maruyama, and Satoru Miyano. Identification of gene regulatory networks by strategic gene disruptions and gene overexpressions. In *SODA '98: Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 695–702, Philadelphia, PA, USA, 1998. Society for Industrial and Applied Mathematics.
- [ITK00] T. Ideker, V. Thorsson, and R Karp. Discovery of regulatory interactions through perturbation: Inference and experimental design. In *Pacific Symposium on Biocomputing 5*, pages 302–313, 2000.
- [KKET03] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *KDD '03: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 137–146, New York, NY, USA, 2003. ACM.
- [KKT05] David Kempe, Jon M. Kleinberg, and Éva Tardos. Influential nodes in a diffusion model for social networks. In *ICALP*, pages 1127–1138, 2005.