Spoofing Linear Cross-Entropy Benchmarking in Shallow Quantum Circuits

Boaz Barak
School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA
`http://boazbarak.org`
b@boazbarak.org

Chi-Ning Chou
School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA
`http://cnchou.github.io`
chiningchou@g.harvard.edu

Xun Gao
Department of Physics, Harvard University, Cambridge, MA, USA
xungao@g.harvard.edu

―――― **Abstract** ――――

The *linear cross-entropy benchmark* (Linear XEB) has been used as a test for procedures simulating quantum circuits. Given a quantum circuit C with n inputs and outputs and purported simulator whose output is distributed according to a distribution p over $\{0,1\}^n$, the linear XEB fidelity of the simulator is $\mathcal{F}_C(p) = 2^n \mathbb{E}_{x \sim p} q_C(x) - 1$, where $q_C(x)$ is the probability that x is output from the distribution $C |0^n\rangle$. A trivial simulator (e.g., the uniform distribution) satisfies $\mathcal{F}_C(p) = 0$, while Google's noisy quantum simulation of a 53-qubit circuit C achieved a fidelity value of $(2.24 \pm 0.21) \times 10^{-3}$ (Arute et. al., Nature'19).

In this work we give a classical randomized algorithm that for a given circuit C of depth d with Haar random 2-qubit gates achieves in expectation a fidelity value of $\Omega(\frac{n}{L} \cdot 15^{-d})$ in running time $\mathsf{poly}(n, 2^L)$. Here L is the size of the *light cone* of C: the maximum number of input bits that each output bit depends on. In particular, we obtain a polynomial-time algorithm that achieves large fidelity of $\omega(1)$ for depth $O(\sqrt{\log n})$ two-dimensional circuits. This is the first such result for two dimensional circuits of super-constant depth. Our results can be considered as an evidence that fooling the linear XEB test might be easier than achieving a full simulation of the quantum circuit.

1 Introduction

Quantum computational supremacy refers to experimental violations of the extended Church Turing Hypothesis using quantum computers. The most famous (and arguably at this point the only) example of such an experiment was carried out by Google [2]. The Google team

constructed a device D that provides a "noisy simulation" of a quantum circuit C with n inputs and n outputs. The device can be thought as a "black box" that samples from a distribution p_D over $\{0,1\}^n$ that (loosely) approximates the distribution q_C that corresponds to measuring C applied to the all-zeroes string 0^n. The quality of the device was measured using a certain benchmark known as the *Linear Cross-Entropy* benchmark (*a.k.a.* Linear XEB). The computational hardness assumption underlying the experiment is that no efficient classical algorithm can achieve a similar score. In this paper we investigate this assumption, giving a new classical algorithm for "spoofing" this benchmark in certain regimes. While our algorithm falls short of spoofing the benchmark in the parameter regime corresponding to the Google experiment, we do manage to achieve non-trivial results for deeper circuits than were known before. To our knowledge, this is the first algorithm that directly targets the linear XEB benchmark, without going through a full simulation of the underlying quantum circuit. Thus our work can be viewed as evidence that obtaining non-trivial performance for this benchmark is *not* equivalent to simulating quantum circuits.

The linear XEB benchmark is defined as follows. Let C be an n-qubit quantum circuit and $q_C : \{0,1\}^n \to [0,1]$ be the pdf of the distribution obtained by measuring $C|0^n\rangle$. For each $x \in \{0,1\}^n$, the instance linear XEB of x is defined as $\mathcal{F}_C(x) := 2^n q_C(x) - 1$. For every probability distribution p, the linear XEB fidelity of p with respect to circuit C is defined as

$$\mathcal{F}_C(p) := \mathop{\mathbb{E}}_{x \sim p} [\mathcal{F}_C(x)] = 2^n \sum_{x \in \{0,1\}^n} q_C(x) p(x) - 1 \,.$$

If C is a fully random circuit, then in expectation a perfect simulation $p = q_C$ achieves $\mathcal{F}_C(p) = 2$.[1] Google's "quantum computational supremacy" experiment demonstrated a noisy simulator sampling from a distribution p with $\mathcal{F}_C(p) \approx (2.24 \pm 0.21) \times 10^{-3}$ for two dimensional 53-qubit circuits of depth 20. A trivial simulation (e.g. a distribution p which is the uniform distribution or another distribution independent of C) will achieve $\mathcal{F}_C(p) = 0$. Motivated by the above, we say that p achieves *non-trivial fidelity* with respect to the circuit C if $\mathcal{F}_C(p) = 1/\mathsf{poly}(n)$.[2]

The computational assumption underlying quantum computational supremacy with respect to some distribution $\mathcal{D}$ over quantum circuits can be defined as follows. For every efficient randomized classical algorithm A, with high probability over $C \sim \mathcal{D}$, if we let A_C be the distribution of A's output on input C, then $\mathcal{F}_C(A_C) = n^{-\omega(1)}$. That is, the distribution output by $A(C)$ has trivial fidelity with respect to C. Aaronson and Gunn [1] showed that this assumption follows from a (very strong) assumption they called "Linear Cross-Entropy Quantum Threshold Assumption" or XQUATH.[3]

In this work, we present an efficient classical algorithm A that satisfies $\mathcal{F}_C(A_C) = \Omega(1)$ for quantum circuit C sampled from a distribution with Haar random 2-qubit gates with small *light cones* (see Definition 4).[4] Specifically, we prove the following theorem:

---

[1] This follows since q_C is the *Porter Thomas distribution*. However, q_C is *not* the maximizer of $\mathcal{F}_C(p)$: a distribution p that has all its mass on the mode x of the distribution q_C will achieve $\mathcal{F}_C(p) \geq \Omega(n)$ for fully random circuits, and even higher values for shallower circuits as we'll see below.

[2] As mentioned above, for an ideal simulation in random circuits the fidelity will be a constant. For noisy quantum circuits such as Google's, the fidelity is roughly $\exp(-\epsilon s)$ where ϵ is the level of noise per gate and $s = \Theta(d \cdot n)$ is the number of gates in the circuit.

[3] While [1] state their result for $\Omega(1)$ fidelity, their proof shows that the XQUATH assumption implies that classical algorithms can not achieve $1/\mathsf{poly}(n)$ empirical fidelity with $poly(n)$ samples.

[4] If C is a quantum circuit and i is an output bit of C, then the *light cone* of i is the set of all input bits j that are connected to i via a path in the circuit. For general circuits the light cone size can be exponential in the depth, but for one or two dimensional circuits, of the type used in quantum supremacy experiment, the light cone size is polynomial in the depth.

▶ **Theorem 1** (Linear XEB for circuits with small light cones). *Let $n, d, L \in \mathbb{N}$ and let $\mathcal{D}$ be a distribution over n-qubit quantum circuits with (i) light cone size at most L, (ii) depth at most d, and (iii) Haar random 2-qubit gates. Then, there exists a classical randomized algorithm A running in $\mathsf{poly}\left(n, 2^L\right)$ time such that*

$$\mathop{\mathbb{E}}_{C \sim \mathcal{D}} [\mathcal{F}_C(A_C)] \geq \left(1 + 15^{-d}\right)^{\left\lfloor \frac{n}{L} \right\rfloor} - 1 \,.$$

For constant dimensional circuits (such as the 2D quantum architecture used by Google), Theorem 1 yields the following corollary:

▶ **Corollary 2** (Constant dimensional circuits). *Let $n \in \mathbb{N}$ and $d = O(\log n)$. Let $c \in \mathbb{N}$ be a constant and $\mathcal{D}$ be the distribution of n-qubit c-dimensional circuits of depth d with Haar random 2-qubit gates. There is a randomized algorithm A running in time $2^{O(d^c)}$ such that*

$$\mathbb{E}_{C \sim \mathcal{D}} \mathcal{F}_C(A_C) = 1/\mathsf{poly}(n) \,.$$

Proof. A c-dimensional of depth d circuit has light-cone of size $L = O(d^c) = n^{o(1)}$ for $d = O(\log n)$. Let $d = \alpha \log n$. By plugging in the parameters of Theorem 1, we see that (using $\log_2 15 < 4$ and $n/L \geq n^{1-o(1)}$) the expected value of the fidelity is at least

$$(1 + 2^{-4d})^{n^{1-o(1)}} - 1 \geq \Omega(\tfrac{n^{1-o(1)}}{n^{4\alpha}}) \,.$$

The right hand side is at least $1/\mathsf{poly}(n)$ for every constant α and in fact is at least $\omega(1)$ for $\alpha < 1/4$. ◀

The bounds of Corollary 2 do *not* correspond to the Google experiment where the depth is roughly comparable to $\sqrt{n}$, rather than logarithmic. However, prior works in the literature were only able to achieve good linear XEB performance for circuits of *constant* depth (see Section 1.2). More importantly (in our view) is that our bounds show that it may be possible to achieve good linear XEB performance without achieving a full simulation.

1.1 From expectation to concentration

In actual experiments, one measures the *empirical* linear XEB, obtained by sampling $x_1, \ldots, x_T$ independently from the distribution p and computing $\frac{1}{T} \sum_{i=1}^{T} 2^n q_C(x_i) - 1$. Thus in our classical simulation we want to go beyond achieving large *expected* linear XEB benchmark, to show that our algorithm A actually achieves non-trivial empirical linear XEB with probability at least inverse polynomial over the choice of the circuit and with a number of samples T that is at most polynomial in n. These probability bounds are more challenging to prove, and at the moment our results are weaker than the optimal bounds one can hope for.

Probability over circuits

For bounding the probability over *circuits* we show in Section 5.1, that in the setting of Theorem 1 , for logarithmic depth circuits, we can obtain $1/\mathsf{poly}(n)$ fidelity with probability at least $1/\mathsf{poly}(n)$. We also obtain more general tradeoffs between the fidelity, probability, and depth, see Corollary 12. We conjecture that random circuits from the distributions we consider exhibit much better concentration, and fact that the fidelity sharply concentrates around its expectation.

Sample complexity, or probability over the algorithms' randomness

Bounding the *sample complexity* of our algorithm is a more difficult task then the expectation analysis because it requires higher moment information on $\mathcal{F}_C(A_C)$. We obtain only partial bounds in this setting, which we believe to be far from optimal. In Section 6 we show that an upper bound for the *collision probability* of q_C is sufficient to give an upper bound for the sample complexity of our algorithm. Specifically, letting $CP(q) = \sum_{x \in \{0,1\}^n} q(x)^2$, we show that if $CP(q) = M \cdot 2^{-n}$ then the number of samples needed for the empirical linear XEB to achieve a value of at least ϵ is $M \cdot \exp\big(O(\epsilon \cdot 15^d)\big)$. In particular, for logarithmic depth circuits we can get inverse-polynomial empirical fidelity using $O(M)$ samples. For *random* quantum circuits, where q_C is the Porter-Thomas distribution (with $q_C(x)$ drawn independently as the square of a mean zero variance 2^{-n} normal variable), it is known that $CP(q_C) = O(2^{-n})$, i.e., $M = O(1)$. For shallow circuits, of the type we study, we show in Lemma 16 that $CP(q_C) = O(2^{-n})$ for random *one dimensional* circuits of depth at least $c \log n$ for some constant $c > 0$, which shows that we can achieve for such circuits inverse polynomial empirical fidelity using a polynomial number of samples. While this is significantly more technically challenging to prove, we conjecture that the same collision probability bound holds for *two dimensional* circuits of depth $\Omega(\sqrt{\log n})$. This conjecture, if true, will imply that for such circuits we can achieve $1/\mathsf{poly}(n)$ empirical fidelity using a polynomial number of samples, and constant fidelity using a sub-exponential (e.g. $\exp\{\exp\{O(\sqrt{\log n})\}\}$) number of samples.[5]

1.2 Prior works

Prior classical algorithms mostly focused on the task of obtaining a full simulation (sampling from $C|0^n\rangle$ or from a distribution close to it in statistical distance). We are not aware of any prior work that directly targeted the linear XEB measure and gave explicit bounds for the performance in this measure that are not implied by approximating the full distribution.

Napp et al [8] gave an algorithm to simulate random two-dimensional circuits of some small constant depth. They gave strong theoretical evidence that up to a certain constant depth, such circuits can be approximated by 1D circuits of small entanglement (i.e., "area law" as opposed to "volume law"), which can be effectively simulated using Matrix Product States [10]. However, [8] also gave evidence that the system undergoes a *phase transition* when the depth is more than some constant size (around 4), at which case the entanglement grows according to a "volume law" and hence their methods cannot be used to simulate circuits of super-constant depth.

Another direction of approximating large quantum circuits has considered the effect of *noise.* Some restricted classes of noisy quantum circuits were shown to be simulated by polynomial time classical algorithms in [4, 11] (in contrast to their noiseless variant [3]). This was also extended to more general random circuits by [6]. Very recent work has given numerical results suggesting that states generated by noisy quantum circuits could be approximated by Matrix Product States or Operators under the state fidelity measure [12, 9].[6] Low degree Fourier expansions yield other candidates for approximating such quantum states [6, 4].

---

[5] Very recently we have learned that Dalzell, Hunter-Jones, and Brandao (personal communication) refuted Conjecture 10 though we have not yet had a chance to verify their proof. The refutation of Conjecture 10 would only rule out a specific approach to upper bound the sample complexity. It is still possible that our algorithm achieves $1/\mathsf{poly}(n)$ empirical fidelity using a polynomial number of samples for *two dimensional* circuits of depth $\Omega(\sqrt{\log n})$.

[6] [12] briefly discusses the linear XEB measure as well, see Figure 7 there.

2 Preliminaries

In this section we introduce some of the notions we use for quantum circuits, and in particular distributions of random quantum circuits of fixed architecture, as well as tensor networks for analyzing quantum circuits. We include this here since some of this notation, and in particular tensor networks, might be unfamiliar to theoretical computer science audience. However, the reader can choose to skip this section and refer back to it as needed.

For $n \in \mathbb{N}$, an n-qubit quantum state $|\psi\rangle = \sum_{x \in \{0,1\}^n} \alpha_x |x\rangle$ is a unit vector in $\mathbb{C}^{2^n}$. We let I, X, Y, Z denote the Pauli matrices where

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \ X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \ Y = \begin{bmatrix} 0 & i \\ -i & 0 \end{bmatrix}, \ \text{and} \ Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

The following definition captures the notion of an "architecture" of a quantum circuit (see Figure 1 for an example):

▶ **Definition 3** (Circuit skeleton and light cone). *Let $n \in 2\mathbb{N}$ and $d \in \mathbb{N}$, an n-qubit depth d circuit skeleton $\mathcal{S}$ is a directed acyclic graph with $d + 2$ layers with the following structure. For convenience, we start the index of layers from 0.*

- *The 0th and the $(d+1)^{th}$ layer has n nodes corresponding to the n input and output qubits. Each node in the first layer has exactly one out-going edge to the next layer while each node the last layer has exactly one in-going edge from the previous layer.*
- *Each of the other layers has exactly $n/2$ nodes and each node has exactly two in-going to the next layer and two out-going edges from the previous layer. Specifically, the first edge i^{th} gate is indexed by $2i - 1$ while the second edge is indexed by $2i$ for each $i = 1, 2, \ldots, n/2$.*
- *For each $i = 1, 2, \ldots, n$, the ith input node connects to the ith edge of the second layer while the ith edge of the $(d+1)^{th}$ layer connects to the ith output node.*

Note that with the above definition, a circuit skeleton $\mathcal{S}$ can be specified by $d + 1$ many permutations $\pi^{(0)}, \pi^{(1)}, \ldots, \pi^{(d)} \in S_n$. Namely, for each $t = 0, 1, 2, \ldots, d - 1$ and $i = 1, 2, \ldots, n$, the ith edge of the tth layer connects to the $\pi^{(t)}(i)^{th}$ edge of the $(t+1)^{th}$ layer.

For every circuit skeleton G, the light cone size *of G is the maximum over all output qubits i of the size of the set $\{j : j$ is input qubit connected to i in $G\}$.*
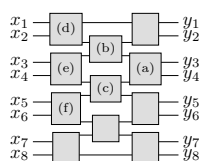
▶ **Figure 1** An example of 1D circuit skeleton with $n = 8$ and $d = 3$. In this example the permutations are $\pi^{(0)} = \pi^{(4)} = \mathsf{id}$, $\pi^{(1)} = \pi^{(3)} = (18765432)$, $\pi^{(2)} = (81234567)$.

Next, we define the light cone for an output qubit and the light cone size for a circuit skeleton.

▶ **Definition 4** (Light cone). *Let $\mathcal{S}$ be a circuit skeleton and i be an output qubit. The light cone of i is the set of all input vertices in $\mathcal{S}$ that has a path from left to right that ends at i. The light cone size of $\mathcal{S}$ is then defined as the largest light cone size of an output qubit in $\mathcal{S}$.*

Note that the light cone size of the 1D circuit in Figure 1 is 6, which is less than the number of qubits. Also, it turns out that computing the marginal of an output qubit only requires the information from the light cone.

▶ **Lemma 5** (Marginal probability and light cone). *Let $\mathcal{S}$ be a circuit skeleton with light cone size L and C be a circuit using skeleton $\mathcal{S}$. For each output qubit of C, the marginal probability can be computed in time $O(2^L)$.*

Proof of Lemma 5. We use the circuit skeleton in Figure 1 as an illustrating example. For an output qubit in C, to compute its marginal probability it suffices to compute the input state to the gate it connects to. For example, for output qubit y_3, it suffices to compute the input state to gate (a).

Similarly, to compute the input state of a gate, it suffices to compute the input states of the gate it connects to from the previous layer. Namely, to compute the input state of gate (a), it suffices to compute that of gate (b) and (c). If we continue this process inductively, the only input state needed to compute the marginal probability of an output bit is then the one lies in its light cone. In this example, to compute the marginal probability of y_3, it suffices to consider only $x_1, x_2, \ldots, x_6$.

Finally, to compute the input states of all the intermediate gates, it suffices to perform $2^L \times 2^L$ matrix vector multiplication because each intermediate state is of size at most L. While all the above operations can be done in $O(2^L)$ times, computing the marginal probability of an output qubits in C only requires $O(2^L)$ time.   ◀

Now, we are able to formally define random quantum circuits.

▶ **Definition 6** (Random quantum circuits). *Let $n \in 2\mathbb{N}$, $d \in \mathbb{N}$. A distribution $\mathcal{D}$ of n-qubit depth d random circuits consists of an n-qubit depth d circuit skeleton $\mathcal{S}$ and ensembles $\{\mathcal{E}_{i,j}\}$ over 4×4 unitary matrices for each $i = 1, \ldots, d$ and $j = 1, \ldots, n/2$.*

A random quantum circuit C sampled from $\mathcal{D}$ by sampling a 4×4 unitary matrix $U_i^{(t)}$ from $\mathcal{E}_i^{(t)}$ and assigning $U_i^{(t)}$ to the ith node of the tth layer for each $t = 1, \ldots, d$ and $i = 1, \ldots, n/2$.

Specifically, if each $\mathcal{E}_i^{(t)}$ is Haar random, then we say $\mathcal{D}$ is Haar random 2-qubit circuits over $\mathcal{S}$.

2.1   Tensor networks

Tensor network is an intuitive graphical language that can be rigorously used in reasoning about multilinear maps. Especially, it finds many applications in quantum computing since the basic operations such as partial measurement are all multilinear maps. In this paper, we restrict our attention to *qubits* (as opposed to the general case of *qudits*) and only to gates that act on two qubits.

(a)   (b)   (c)
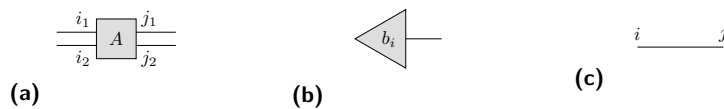
🟨 **Figure 2** Three basic elements in tensor networks.   (a) Gate: the figure represents $\sum_{b_i, b_j \in \{0,1\}} A_{b_i, b_j} |b_i\rangle \langle b_j|$. (b) State: the figure represents $\langle b_i|$. (c) Line: the figure represents δ_{b_i, b_j}.

In a tensor network, we represent a unitary matrix (e.g., a gate) as a box with lines on the sides (see Figure 2a). Each line represents a coordinate of the gate and in this paper each coordinate has dimension 2 and is indexed by $\{0, 1\}$. Specifically, a line on the left represents a column vector (i.e., $|\cdot\rangle$) while a line on the right represents a row vector (i.e., $\langle\cdot|$).[7] For example, Figure 2a represents $\sum_{b_i, b_j \in \{0,1\}} A_{b_i, b_j} |b_i\rangle \langle b_j|$.

---

[7] This is when the tensor network is written left to right - sometimes it is written top to bottom, in which case a line on the top represents a column vector and a line on the bottom represents a row vector.

Similarly, a state (e.g., a qubit)is represented by a triangle with line only on one side and it is a $|\cdot\rangle$ (resp. $\langle\cdot|$) if the free-end of the line is left (resp. right). For example, Figure 2b represents $\langle b_i|$.

Semantically, a pure line refers to an indicator function[8] for its two ends. For example, the line in Figure 2c reads as $\sum_{b)i,b_j\in\{0,1\}} \delta_{b_i,b_j} \langle b_i|b_j\rangle$ where $\delta_{b_i,b_j} = 1$ if $b_i = b_j$; otherwise it is 0.

3   Our Algorithm

We now describe our classical algorithm that spoofs the linear cross-entropy benchmark in shallow quantum circuits. The key idea is that rather than directly simulating the whole quantum circuit, our algorithm only computes the marginal distributions of few output qubits and then samples substrings for those qubit accordingly. We sample the remaining subits uniformly at random. Intuitively, due to the correlation on those output qubits, one can expect that the linear cross-entropy of our algorithm could be better than uniform distribution, but the analysis is somewhat delicate. Because consider shallow quantum circuits (of at most logarithmic light cone size), the marginal of few output qubits can be efficiently computed.

■ **Algorithm 1** Classical algorithm for spoofing linear XEB in shallow quantum circuits.

Input: A quantum circuit C sampled from $\mathcal{D}_{\mathcal{S}}$, a Haar random distribution over an n-qubit circuit skeleton $\mathcal{S}$ with light cone size at most L.

1: We set m be some parameter in $\{1, \ldots, \lfloor n/L \rfloor\}$. (We set $m = \lfloor n/L \rfloor$ to obtain the result of Theorem 1 as stated.)
2: Find m output qubits $i_1, \ldots, i_m$ such that their light cones are disjoint.
3: Calculate the marginal probability of each output qubits $i_1, \ldots, i_m$.
4: Sample $x_{i_1}, \ldots, x_{i_m}$ according to the marginal probabilities calculated in the previous step. For any $i \notin \{i_1, \ldots, i_m\}$, sample x_i uniformly random from $\{0,1\}$.

Output: x.

Running time of the algorithm

The total running time of Algorithm 1 is at most $\mathsf{poly}(n, 2^L)$. Finding m outputs with disjoint light cones takes $\mathsf{poly}(n)$ time by a greedy algorithm. The second step takes $\mathsf{poly}\left(n, 2^L\right)$ time because it suffices to keep track of the $2^L \times 2^L$ density matrix recording the marginal probability of every qubit in the light cone of i_j for each $j \in [m]$ (see Lemma 5). The final step of sampling uniform bits for the remaining outputs can be done in polynomial time.

3.1   Analysis

The following theorem implies Theorem 1 by setting $m = \lfloor n/L \rfloor$:

▶ **Theorem 7** (Linear XEB for circuits with small light cones.)**.** *Let $n, d, L \in \mathbb{N}$ and let $\mathcal{D}$ be a distribution over n-qubit quantum circuits with (i) light cone size at most L, (ii) depth at most d, and (iii) Haar random 2-qubit gates. Then, letting A_C be the distribution output by Algorithm 1 on input C,*

$$\underset{C\sim\mathcal{D}}{\mathbb{E}} [\mathcal{F}_C(A_C)] \geq \left(1 + 15^{-d}\right)^m - 1 \ ,$$

where m is the parameter chosen in step 1 of the algorithm.

---

[8] Also known as *contraction*.

The proof of Theorem 7 consists of three steps:

1. We reduce analyzing analyzing the expectation when the algorithms samples the marginals of m output qubits into analyzing it for a single output qubit.
2. We apply the integration formula for Haar measure and rewrite the expected linear XEB of a single qubit into a tensor network.
3. We then perform a change of basis on the tensor network and turn the single qubit analysis into a Markov chain problem where the expected linear XEB of a single qubit can be easily lower bounded.

Since the heart of the proof is the single output qubit analysis, we will describe it first.

4 Single qubit analysis

In this section, we prove the $m = 1$ case of our algorithm. That is, we prove that for a single output qubit, the expected contribution to linear XEB is of the order of 15^{-d}.

▶ **Theorem 8** (Linear XEB of a single output qubit). *Let $n, d \in \mathbb{N}$ and $\mathcal{D}$ be distribution over n-qubit quantum circuits with depth at most d and with Haar random 2-qubit gates. For $C \sim \mathcal{D}$, let U denote the unitary matrix computed by C. For each $i \in [n]$, we have*

$$\mathbb{E}_{C \sim \mathcal{D}} \left[q_{C,i,0}^2 + q_{C,i,1}^2 \right] \geq \frac{1 + 15^{-d}}{2},$$

where $q_{C,i,b} = \Pr_{x \sim q_C}[x_i = b]$.

We prove Theorem 8 by reducing to a Markov chain problem using tensor networks. Without loss of generality we can assume $i = 1$. Observe that

$$q_{C,1,0}^2 + q_{C,1,1}^2 = \frac{1 + \mathrm{tr}\left(Z \otimes I^{\otimes n-1} U^\dagger |0^n\rangle \langle 0^n| U\right)^2}{2} \tag{1}$$

So our goal is to show that

$$\mathbb{E}_{C \sim \mathcal{D}} \left[\mathrm{tr}\left(Z \otimes I^{\otimes n-1} U^\dagger |0^n\rangle \langle 0^n| U\right)^2 \right] \geq 15^{-d} \quad . \tag{2}$$

Let us start with rewriting the trace term of Equation 2 into an equivalent tensor network expression as follows.

$$\mathrm{tr}\left(Z \otimes I^{\otimes n-1} U^\dagger |0^n\rangle \langle 0^n| U\right)^2 = \quad$$

Next, for a single gate g in a quantum circuit, its expected behavior over the choice of 2-qubit Haar random gates can be characterized in the following lemma.

▶ **Lemma 9.** *Let U_g be Haar random 2-qubit gate, then the following holds.*

where

$$
M_{\sigma_1,\sigma_2,\sigma_1',\sigma_2'} =
\begin{array}{c}
\begin{array}{ccccc} II & IX & IY & \cdots & ZZ \end{array} \\
\left(\begin{array}{ccccc}
1 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{15} & \frac{1}{15} & \cdots & \frac{1}{15} \\
0 & \frac{1}{15} & \frac{1}{15} & \cdots & \frac{1}{15} \\
0 & \vdots & \vdots & \ddots & \vdots \\
0 & \frac{1}{15} & \frac{1}{15} & \cdots & \frac{1}{15}
\end{array}\right)
\begin{array}{c} II \\ IX \\ IY \\ \vdots \\ ZZ \end{array}
\end{array} \ .
$$

The proof of Lemma 9 is based on the integration formula [5] for Haar measure. We postpone the proof of Lemma 9 to Subsection 4.1. Intuitively, the lemma says that by a change a basis, the expected behavior of a single Haar random 2-qubit gate can be exactly understood by an explicit transition matrix M. By the linearity of taking expectation, we can apply Lemma 9 on every gates in the circuit C and thus the whole tensor network is simplified to a *Markov chain*. Concretely, we have the following lemma.

▶ **Lemma 10** (Rewrite Equation 2 as a Markov chain). *Let $n, d \in \mathbb{N}$ and $\mathcal{D}$ be a Haar random distribution over an n-qubit depth d circuit skeleton $\mathcal{S}$ with permutations $\pi^{(0)}, \pi^{(1)}, \ldots, \pi^{(d)}$. For $C \sim \mathcal{D}$, let U denote the unitary matrix computed by C.*

$$
\mathop{\mathbb{E}}_{C \sim \mathcal{D}} \left[\mathrm{tr}\left(Z \otimes I^{\otimes n-1} U^\dagger |0^n\rangle \langle 0^n| U \right)^2 \right] = \sum_{\substack{\sigma_{i'}^{(t')} \in \{I, X, Y, Z\} \\ i'=1,2,\ldots,n \\ t'=1,2,\ldots,d+1}} \prod_{t=0}^{d+1} V^{(t)}\left(\left\{ \sigma_{i'}^{(t')} \right\} \right) \tag{3}
$$

where

$$
V^{(t)}\left(\left\{ \sigma_{i'}^{(t')} \right\} \right) = \begin{cases}
\prod_{i=1}^{n} \mathrm{tr}\left(\frac{I+Z}{2} \frac{\sigma_i^{(1)}}{\sqrt{2}} \right)^2 & , \ \text{if } t = 0 \\
\prod_{i=1}^{n/2} M_{\sigma_{2i-1}^{(t)}, \sigma_{2i}^{(t)}, \sigma_{\pi^{(t)}(2i-1)}^{(t+1)}, \sigma_{\pi^{(t)}(2i)}^{(t+1)}} & , \ \text{if } t = 1, 2, \ldots, d \\
\mathrm{tr}\left(\frac{\sigma_1^{(d+1)}}{\sqrt{2}} Z \right)^2 \cdot \prod_{i=2}^{n} \mathrm{tr}\left(\frac{\sigma_i^{(d+1)}}{\sqrt{2}} I \right)^2 & , \ \text{if } t = d+1 \, .
\end{cases}
$$

The proof of Lemma 10 is based on a careful composition of applying Lemma 9 on each of the gates. We postpone the proof of Lemma 10 to Subsection 4.2. Now, we are ready to prove Theorem 8 and complete the analysis for the expected linear XEB of single output qubit.

Proof of Theorem 8. Lemma 10 rewrites the desiring quantity into the form of a Markov chain so now it suffices to show that the right hand side of Equation 3 is at least 15^{-d}.

Notice that for every possible assignment to $\{\sigma_{i'}^{(t')}\}$, $V(\{\sigma_{i'}^{(t')}\}) \geq 0$. That is, it suffices to find an assignment such that $\prod_{t=0}^{d+1} V^{(t)}(\{\sigma_{i'}^{(t')}\}) \geq 15^{-d}$. Specifically, let us consider the following assignment. For all $i = 1, 2, \ldots, n$ and $t = 1, 2, \ldots, d+1$, let

$$
\sigma_i^{(t)} = \begin{cases}
Z & , \ \text{if } \pi^{(d)} \circ \pi^{(d-1)} \circ \cdots \circ \pi^{(t)}(i) = 1 \\
I & , \ \text{else} \, .
\end{cases}
$$

To analyze this assignment, let us start with the last layer. There we have $\sigma_1^{(d+1)} = Z$ and $\sigma_i^{(d+1)} = I$ for each $i = 2, 3, \ldots, n$ and thus

$$V^{(d+1)}(\{\sigma_{i'}^{(t')}\}) = \text{tr}\left(\frac{\sigma_1^{(d+1)}}{\sqrt{2}}Z\right)^2 \cdot \prod_{i=2}^n \text{tr}\left(\frac{\sigma_i^{(d+1)}}{\sqrt{2}}I\right)^2$$

$$= \text{tr}\left(\frac{ZZ}{\sqrt{2}}\right)^2 \cdot \text{tr}\left(\frac{II}{\sqrt{2}}\right)^{2(n-1)} = 2^n .$$

Next, for each $t = 1, 2, \ldots, d$ and $i = 1, 2, \ldots, n$, observe that $\sigma_i^{(t)} = \sigma_{\pi^{(t+1)}(i)}^{(t)}$ due to the choice of the assignment. As a result, all the $M_{\sigma_{2i-1}^{(t)}, \sigma_{2i}^{(t)}, \sigma_{\pi^{(t)}(2i-1)}^{(t+1)}, \sigma_{\pi^{(t)}(2i)}^{(t+1)}}$ will be either $M_{I,I,I,I}$ or $M_{I,Z,I,Z}$. Specifically, for each $t = 1, 2, \ldots, d$, since there is exactly one Z appears among $\{\sigma_i^{(t)}\}_{i=1,\ldots,n}$ while the rest are Is, there is also exactly one $M_{I,Z,I,Z}$ term contributes in $V^{(t)}(\{\sigma_{i'}^{(t')}\})$ while the other terms are $M_{I,I,I,I}$. Namely, we have

$$V^{(t)}(\{\sigma_{i'}^{(t')}\}) = M_{I,Z,I,Z} \cdot (M_{I,I,I,I})^{n/2-1} = \frac{1}{15}$$

for each $t = 1, 2, \ldots, d$.

Finally, since there is exactly one Z appears in $\{\sigma_i^{(0)}\}_{i=1,\ldots,n}$ while the rest are Is, we have

$$V^{(0)}(\{\sigma_{i'}^{(t')}\}) = \prod_{i=1}^n \text{tr}\left(\frac{I+Z}{2}\frac{\sigma_i^{(0)}}{\sqrt{2}}\right)^2 = \text{tr}\left(\frac{I+Z}{2}\frac{Z}{\sqrt{2}}\right)^2 \cdot \text{tr}\left(\frac{I+Z}{2}\frac{I}{\sqrt{2}}\right)^{2(n-1)}$$

$$= \left(\frac{1}{\sqrt{2}}\right) \cdot \left(\frac{1}{\sqrt{2}}\right)^{2(n-1)} = \frac{1}{2^n} .$$

To sum up, we conclude that $V(\{\sigma_{i'}^{(t')}\}) = \prod_{t=0}^{d+1} V^{(t)}(\{\sigma_{i'}^{(t')}\}) = 15^{-d}$ as desired. Specifically, this implies Equation 2, i.e., $\mathbb{E}_{C \sim \mathcal{D}}\left[\text{tr}\left(Z \otimes I^{\otimes n-1} U^\dagger |0^n\rangle \langle 0^n| U\right)^2\right] \geq 15^{-d}$. Combine with Equation 1, this completes the proof of Theorem 8. ◀

4.1 Proof of Lemma 9

We start with applying the integration formula for Haar random matrix and considering its tensor netowrok representation.

▶ **Lemma 11** ([5, Equation 2.4]). *Let U be a Haar random 2-qubit gate, then we have the following. For each $x_a, x_b, x_c, x_d, y_a, y_b, y_c, y_d \in \{0,1\}^2$,*

$$\mathbb{E}_U\left[U_{x_a y_a}U_{x_b y_b}^\dagger U_{x_c y_c}U_{x_d y_d}^\dagger\right] = \frac{1}{15} \cdot \left[\delta_{x_a x_b}\delta_{x_c x_d}\delta_{y_a y_b}\delta_{y_c y_d} + \delta_{x_a x_d}\delta_{x_b x_c}\delta_{y_a y_d}\delta_{y_b y_c}\right]$$

$$- \frac{1}{60} \cdot \left[\delta_{x_a x_b}\delta_{x_c x_d}\delta_{y_a y_d}\delta_{y_b y_c} + \delta_{x_a x_d}\delta_{x_b x_c}\delta_{y_a y_b}\delta_{y_c y_d}\right] .$$

The above equation can be represented as the following tensor network.

Next, the idea is to apply the Pauli identity (i.e., on each pair of $(1_a, 1_b), (1_c, 1_d), (2_a, 2_b),$ $(2_c, 2_d), (1'_a, 1'_b), (1'_c, 1'_d), (2'_a, 2'_b),$ and $(2'_c, 2'_d)$. Intuitively, this is doing a change of basis from the standard basis to Pauli basis.

Let us first apply the Pauli identity on $(1_a, 1_b)$ and $(1_c, 1_d)$ note that we have

$$(4)$$

and

$$(5)$$

That is, the tensor network is non-zero only if $\sigma = \sigma'$. Thus, we only need one variable $\sigma_1 \in \{I, X, Y, Z\}$ to handle $(1_a, 1_b)$ and $(1_c, 1_d)$. Similarly, we can use $\sigma_2, \widetilde{\sigma_1}, \widetilde{\sigma_2} \in \{I, X, Y, Z\}$ to handle other pairs respectively. The equation becomes the following.

To finish the proof of Lemma 9, we have to explicitly calculate the value of the tensor network for each choice of $\sigma_1, \sigma_2, \sigma_{1'}, \sigma_{2'} \in \{I, X, Y, Z\}$. Again, by Equation 4 and Equation 5, we have the following observations.

- If $\sigma_1 = \sigma_2 = \sigma_{1'} = \sigma_{2'} = I$, then the value is $2^{-8} \cdot \left(15^{-1} \cdot (2^8 + 2^4) - 60^{-1} \cdot (2^6 + 2^6)\right) = 2^{-4}$.
- If $\sigma_1 = \sigma_2 = I$ and at least one of $\sigma_{1'}, \sigma_{2'}$ is not I, or at least one of σ_1, σ_2 is not I and $\sigma_{1'} = \sigma_{2'} = I$, then the value is $2^{-4} \cdot \left(15^{-1} \cdot (0 + 2^4) + 60^{-1} \cdot (2^6 + 0)\right) = 0$.
- For all the other cases, the value is $2^{-4} \cdot \left(15^{-1} \cdot (0 + 2^4) + 60^{-1} \cdot (0 + 0)\right) = 2^{-4} \cdot 15^{-1}$.

Finally, we take out the 2^{-4} and evenly distribute it to the Pauli gates outside. Namely, each of them gets an extra $1/\sqrt{2}$ factor as shown in the equation. This completes the proof of Lemma 9.

4.2 Proof of Lemma 10

Let us do a change of basis from the standard basis to the Pauli basis. Concretely, we apply Lemma 9 on every gate. Note that by the independence of each gate and the linearity of expectation, the tth layer of the circuit becomes the following for each $t = 1, 2, \ldots, d$.

Next, the ith output wire at the tth layer, i.e., the wires indexed by $\widetilde{i}_a, \widetilde{i}_b, \widetilde{i}_c, \widetilde{i}_d$, connects to the $(\pi^{(t)}(i))^{\text{th}}$ input wire at the $(t+1)^{\text{th}}$ later, i.e., the wires indexed by $\pi^{(t)}(i)_a, \pi^{(t)}(i)_b, \pi^{(t)}(i)_c, \pi^{(t)}(i)_d$. By the orthogonality of Pauli gates (i.e., we have $\text{tr}\left(\dfrac{\widetilde{\sigma}_i^{(t)}}{\sqrt{2}} \dfrac{\sigma^{(t+1)}_{\pi^{(t)}(i)}}{\sqrt{2}} \right) = \delta_{\widetilde{\sigma}_i^{(t)}, \sigma^{(t+1)}_{\pi^{(t)}(i)}}$ for all $i = 1, 2, \ldots, n$. To sum up, the 1st to d^{th} layer is equivalent to following.

Finally, let us plug in the input and output layer. Recall that the input layer contains 4 copies of $|0^n\rangle$ and the output layer contains 2 copies of $Z \otimes I^{\otimes n-1}$. Concretely, the contribution from the input layer would be

$$\prod_{i=1}^{n} \left(\langle 0| \frac{\sigma_1^{(1)}}{\sqrt{2}} |1\rangle \right)^2 = \prod_{i=1}^{n} \text{tr}\left(\frac{I+Z}{2} \frac{\sigma_i^{(1)}}{\sqrt{2}} \right)^2$$

while the contribution from the output layer would be

$$\text{tr}\left(\frac{\sigma_1^{(d+1)}}{\sqrt{2}} Z \right)^2 \cdot \prod_{i=2}^{n} \text{tr}\left(\frac{\sigma_i^{(d+1)}}{\sqrt{2}} I \right)^2 .$$

This completes the proof of Lemma 10.

5 Wrapping up: from single output bit to many bits

In this section we complete the proof of Theorem 1.

We will use the following notation. Let $q(x)$ be a pdf over $x \in \{0,1\}^n$. For any $I \subset [n]$ and $x_I \in \{0,1\}^I$, let $q(I, x_I)$ denote the marginal probability of the output qubit at location I being x_I. Formally, $q(I, x_I) = \sum_{\substack{y \in \{0,1\}^n \\ y_I = x_I}} q(y)$. For a fixed input C, let $I = \{i_1, \ldots, i_m\}$ be the output qubits selected by Algorithm 1. Note that Algorithm 1 will choose the same I for each C sampled from $\mathcal{D}$. By the design of Algorithm 1, $A_C(x) = \frac{1}{2^{n-m}} q_C(I, x_I)$ for every $x \in \{0,1\}^n$. Thus, the linear XEB of A_C is the following.

$$\mathcal{F}_C(A_C) = 2^n \sum_{x \in \{0,1\}^n} q_C(x) A_C(x) - 1 = 2^n \sum_{x \in \{0,1\}^n} q_C(x) \frac{q_C(I, x_I)}{2^{n-m}} - 1 \qquad (6)$$

$$= 2^m \sum_{x_I \in \{0,1\}^I} q_C(I, x_I)^2 - 1 .$$

Note that because their light cones are disjoint, we have $q_C(I, x_I) = \prod_{j=1}^{m} q_C \left(\{i_j\}, x_{i_j} \right)$. Thus, the equation becomes

$$= \sum_{x_I \in \{0,1\}^I} \prod_{j=1}^{m} 2 q_C(\{x_{i_j}\}, x_{i_j})^2 - 1. \tag{7}$$

Now, let us take expectation on the linear XEB over $\mathcal{D}$. Since $\mathcal{D}$ fixes the structure of the circuit and the randomness only lies in the choice of gates, $q_C \left(\{i_j\}, x_{i_j} \right)$ is independent to each other. Namely,

$$\mathop{\mathbb{E}}_{C \sim \mathcal{D}} [\mathcal{F}_C(A_C)] = \sum_{x_I \in \{0,1\}^I} \prod_{j=1}^{m} 2 \mathop{\mathbb{E}}_{C \sim \mathcal{D}} \left[q_C(\{x_{i_j}\}, x_{i_j})^2 \right] - 1$$

$$= \prod_{j=1}^{m} 2 \mathop{\mathbb{E}}_{C \sim \mathcal{D}} \left[q_C(\{x_{i_j}\}, 0)^2 + q_C(\{x_{i_j}\}, 1)^2 \right] - 1. \tag{8}$$

Using the single qubit analysis (Theorem 8), we can complete the proof of Theorem 1 as follows.

Proof of Theorem 1. Apply Theorem 8 on Equation 8, we have

$$\mathop{\mathbb{E}}_{C \sim \mathcal{D}} [\mathcal{F}_C(A_C)] \geq \left(1 + 15^{-d} \right)^m - 1$$

as desired. ◀

5.1 Probability over circuits

Using Theorem 1, we can obtain the following lower bound on the probability over the choice of the circuit C of obtaining non-trivial fidelity:

▶ **Corollary 12** (Lower bounding for the probability of success). *Let $n, d, \mathcal{D}, L$ be as in Theorem 1. Then there is a randomized $poly(n, 2^L)$ time algorithm such that for every $1 \leq n \leq \left\lfloor \frac{n}{L} \right\rfloor$ and $0 < \epsilon < 1$,*

$$\mathop{\Pr}_{C \sim \mathcal{D}} \left[\mathcal{F}_C(A_C) \geq \left(\left(1 + 15^{-d} \right)^m - 1 \right) \right] \geq \frac{(1 - \epsilon) \cdot \left(\left(1 + 15^{-d} \right)^m - 1 \right)}{2^m - 1} \geq \Omega \left(\frac{m \cdot 15^{-d}}{2^m} \right).$$

Proof of Corollary 12. The idea is simple - since our algorithm picks $n - m$ bits uniformly at random, for every circuit C, by Equation 7, $\mathcal{F}_C(A_C) \leq 2^m$. Now, for any $0 < \epsilon < 1$, let $\delta = \Pr_{C \sim \mathcal{D}} \left[\mathcal{F}_C(A_C) > \epsilon \cdot \left(\left(1 + 15^{-d} \right)^m - 1 \right) \right]$, we have

$$\left(1 + 15^{-d} \right)^m - 1 \leq \mathop{\mathbb{E}}_{C \sim \mathcal{D}} [\mathcal{F}_C(A_C)] \leq \delta \cdot 2^m + \epsilon \cdot \left(\left(1 + 15^{-d} \right)^m - 1 \right).$$

Thus,

$$\delta \geq \frac{(1 - \epsilon) \cdot \left(\left(1 + 15^{-d} \right)^m - 1 \right)}{2^m}. \qquad \blacktriangleleft$$

6 Sample complexity analysis

In this section, we discuss the empirical linear XEB of our algorithm. Namely, how many samples are required so that the empirical average of the linear XEB can be non-trivially lower bounded. Specifically, the goal would be the following. For some $T = \mathsf{poly}(n)$,

$$\Pr_{\substack{C \sim \mathcal{D} \\ x_1, \ldots, x_T \sim A_C}} \left[\frac{1}{T} \sum_{i=1}^{n} \mathcal{F}_C(x_i) = \Omega(1) \right] \geq \frac{1}{\mathsf{poly}(n)} \,. \tag{9}$$

In Section 5, we have shown that the expectation of the linear XEB of our algorithm is at least $\left(1 + 15^{-d}\right)^m$ for $1 \leq m \leq \lfloor n/L \rfloor$ with probability $1/\mathsf{poly}(n)$ over the choice of random circuits. Thus, to achieve Equation 9, it suffices to show that the probability of the empirical average of the linear XEB deviating from $\mathcal{F}_C(x)$ is small.

In general, it is a difficult task to rigorously upper bound the sample complexity of linear XEB. The reason is that such analysis needs to handle higher moment of q_C which is highly non-trivial for even 2D circuits. In this work we stick with the simpler case of analyzing the variance of linear XEB in Lemma 13. We further show in Lemma 14 that an inverse exponential bound on the collision probability of q_C would be sufficient for giving $\mathsf{poly}(n)$ upper bound for the sample complexity.

6.1 A variance/collision probability approach

The variance of $\mathcal{F}_C(x)$ is sufficient for upper bounding the number of samples required for the empirical linear XEB to converge. Specifically, Chebyshev's inequality implies that with $\mathsf{Var}_{x \sim p}[\mathcal{F}_C(x)]/(\epsilon^2 \delta)$ many samples, the empirical XEB is at least $\mathcal{F}_C(p) - \epsilon$ with probability δ over the randomness of p. Note that here the circuit C is fixed.

▶ **Lemma 13.** *Let C be an n-qubit quantum circuit and q_C be the pdf of the distribution obtained from $C\,|0^n\rangle$. For any pdf $p : \{0,1\}^n \to [0,1]$ and $\epsilon, \delta \in (0,1)$, we have*

$$\Pr_{x_1, \ldots, x_T \sim p} \left[\frac{1}{T} \sum_{i=1}^{T} \mathcal{F}_C(x_i) \leq \mathcal{F}_C(p) - \epsilon \right] \leq \delta$$

when $T \geq \frac{\mathsf{Var}_{x \sim p}[\mathcal{F}_C(x)]}{\epsilon^2 \delta}$.

Proof. Since $\{\mathcal{F}_C(x_i)\}$ are i.i.d. random variables with mean $\mathcal{F}_C(p)$ and variance $\mathsf{Var}_{x \sim p}[\mathcal{F}_C(x)]$, by Chebyshev's inequality, we have

$$\Pr_{x_1, \ldots, x_T \sim p} \left[\frac{1}{T} \sum_{i=1}^{T} \mathcal{F}_C(x_i) < \mathcal{F}_C(p) - \epsilon \right] \leq \frac{\mathsf{Var}_{x \sim p}[\mathcal{F}_C(x)]}{T \cdot \epsilon^2} \,.$$

As we pick $T \geq \frac{\mathsf{Var}_{x \sim p}[\mathcal{F}_C(x)]}{\epsilon^2 \delta}$, the above error is at most δ desired. ◄

The following lemma further shows that to upper bound the variance of our algorithm, it suffices to bound the *collision probability* of the ideal distribution.

▶ **Lemma 14.** *Let $n, d, L \in \mathbb{N}$ and $\mathcal{D}$ be distribution over n-qubit quantum circuits with (i) light cone size at most L, (ii) depth at most d, and (iii) with Haar random 2-qubit gates. Let $1 \leq m \leq \lfloor n/L \rfloor$ and A be the algorithm from Algorithm 1, we have*

$$\mathsf{Var}_{x \sim A_C} [\mathcal{F}_C(x)] \leq 2^{m+n} \sum_{x \in \{0,1\}^n} q_C(x)^2$$

where $\sum_{x \in \{0,1\}^n} q_C(x)^2$ is also known as the collision probability of q_C.

Proof of Lemma 14. Consider the variance of the linear XEB of our algorithm A_C as follows.

$$\operatorname*{Var}_{x \sim A_C} [\mathcal{F}_C(x)] \leq \operatorname*{\mathbb{E}}_{x \sim A_C} [2^{2n} q_C(x)^2] = 2^{2n} \sum_{x \in \{0,1\}^n} A_C(x) q_C(x)^2 \,.$$

Recall that $A_C(x) \leq 2^{m-n}$ for all x, thus the equation becomes

$$\leq 2^{m+n} \sum_{x \in \{0,1\}^n} q_C(x)^2 \,. \qquad \blacktriangleleft$$

To have some intuition on Lemma 14, the right hand side is minimized when q_C is the uniform distribution over $\{0,1\}^n$ where the collision probability is 2^{-n}. In such case, the variance of our algorithm is $O(2^m)$. When choosing $m = O(\log n)$, the sample complexity of our algorithm would be $\mathsf{poly}(n)$ as desired.

In general, using the variance/collision probability to upper bound the sample complexity might not be tight. For example, consider the distribution of a sequence of independent biased coins, i.e., $q(x) = (1/2+\epsilon)^{\|x\|_1} \cdot (1/2-\epsilon)^{n-\|x\|_1}$ for each $x \in \{0,1\}^n$. Then the variance $\mathsf{Var}_{x \sim q}[q(x)]$ is exponentially large, however, the sample complexity of having the empirical average of $q(x)$ being of the order of $\mathbb{E}_{x \sim q}[q(x)]$ is $O(1)$ with high probability. Specifically, when the depth of the random circuit is 1, then the marginal distribution looks like the above biased coins distribution with high probability and thus undesirable.

On the other extreme where the random circuit is very deep, it is known that the marginal distribution will converge to the *Porter Thomas distribution* and its collision probability is $O(2^{-n})$ [2].

In Subsection 6.2, we further show that the collision probability of q_C is $O(2^{-n})$ in expectation for 1D circuit of depth at least $(\log n)/\log(5/4)$. While the proof could potentially be extended to 2D circuit and beyond, we leave it as a future direction and state the following conjecture.

▶ **Conjecture 10.** *Let $n, d \in \mathbb{N}$ and $\mathcal{D}$ be a distribution of n-qubit. For a circuit C, denote q_C as the pdf of $C\ket{0^n}$. We conjecture that there exists a constant $c > 0$ such that when $\mathcal{D}$ is the distribution over 2D random circuits of depth $d \geq c\sqrt{\log n}$,*

$$\operatorname*{\mathbb{E}}_{C \sim \mathcal{D}} \left[\sum_{x \in \{0,1\}^n} q_C(x)^2 \right] = O\left(\frac{1}{2^n}\right) \,.$$

6.2   The sample complexity of 1D random circuits of logarithmic depth

In this subsection, we formally prove that the sample complexity of our algorithm is $O\left(2^m\right)$ for random 1D circuits with high probability.

▶ **Theorem 15.** *Let $n \in \mathbb{N}$ and $\mathcal{D}$ be the distribution over n-qubit 1D quantum circuits with depth $d = \Omega(\log n)$ and with Haar random 2-qudit gates where the dimension of the qudit is at least 4. Let $1 \leq m \leq \lfloor n/2d \rfloor$ be the number of output qubits used by our algorithm. Then for any $\delta \in (0,1)$, we have*

$$\operatorname*{Pr}_{C \sim \mathcal{D}} \left[\operatorname*{Var}_{x \sim A_C} [\mathcal{F}_C(x)] = O\left(\frac{2^m}{\delta}\right) \right] \geq 1 - \delta \,.$$

Specifically, combine with Theorem 1, we have

$$\operatorname*{Pr}_{\substack{C \sim \mathcal{D} \\ x_1,\dots,x_T \sim A_C}} \left[\sum_{i=1}^T \mathcal{F}_C(x_i) = \Omega\left(\frac{1}{\mathsf{poly}(n)}\right) \right] \geq \frac{1}{\mathsf{poly}(n)}$$

when $T = \Omega(1)$.

Proof of Theorem 15. Let us first show that upper bounding the second moment of q_C is sufficient for proving Theorem 15. Consider the variance of the linear XEB of our algorithm A_C as follows.

$$\operatorname*{Var}_{x \sim A_C}[\mathcal{F}_C(x)] \leq \operatorname*{\mathbb{E}}_{x \sim A_C}[2^{2n} q_C(x)^2] = 2^{2n} \sum_{x \in \{0,1\}^n} A_C(x) q_C(x)^2 .$$

Recall that $A_C(x) \leq 2^{m-n}$ for all x, thus the equation becomes

$$\leq 2^{m+n} \sum_{x \in \{0,1\}^n} q_C(x)^2 .$$

Next, the lemma below shows that the second moment term $\sum_{x \in \{0,1\}^n} q_C(x)^2$ is exponentially small with high probability over the choice of C.

▶ **Lemma 16.** *Let $n \in \mathbb{N}$ and $\mathcal{D}$ be the distribution over n-qubit 1D quantum circuits with depth at least $\frac{\log n}{\log(5/4)}$ and with Haar random 2-qubit gates. Then we have*

$$\operatorname*{\mathbb{E}}_{C \sim \mathcal{D}} \left[\sum_{x \in \{0,1\}^n} q_C(x)^2 \right] = O\left(\frac{1}{2^n} \right) .$$

The proof of Lemma 16 is based on the Ising model analysis by [7]. We postpone it to Subsection 6.3. Now, let us complete the proof of Theorem 15. By Lemma 16, we have

$$\operatorname*{\mathbb{E}}_{C \sim \mathcal{D}} \left[\operatorname*{Var}_{x \sim A_C}[\mathcal{F}_C(x)] \right] \leq 2^{m+n} \operatorname*{\mathbb{E}}_{C \sim \mathcal{D}} \left[q_C(x)^2 \right] \leq O\left(2^m \right) .$$

Thus, for any $\delta > 0$, by Markov's inequality, we have $\Pr_{C \sim \mathcal{D}}[\operatorname{Var}_{x \sim A_C}[\mathcal{F}_C(x)] = O(2^m / \delta)] \geq 1 - \delta$ as desired.   ◀

6.3   Proof of Lemma 16

It turns out that the previous Markov chain approach in analysis the expected linear XEB of our algorithm is not sufficient for upper bounding the expectation of $\sum_{x \in \{0,1\}^n} q_C(x)^2$. We thus consider a different approach by reducing the quantity to a combinatorial problem in a spin system on lattice. The proof is highly inspired by a recent paper of Hunter [7].

For the convenience of the analysis, here we fix the following skeleton for 1D circuit while the result can be easily extended to other variants.
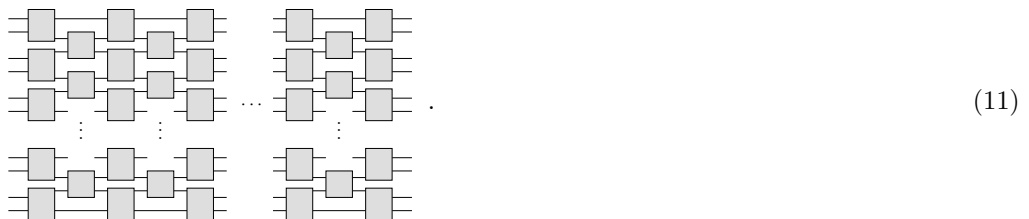
(11)

Step 1: Reducing to counting spin configurations on a hexagonal lattice

First, let us rewrite $\sum_{x\in\{0,1\}^n} q_C(x)^2$ into an equivalent tensor network.

$$\sum_{x\in\{0,1\}^n} q_C(x)^2 = \sum_{x\in\{0,1\}^n} \mathrm{tr}\left(\langle 0^n| U |x\rangle \langle x| U^\dagger |0^n\rangle\right)^2$$

In Lemma 9, we change the basis to the Pauli basis and replace the expectation of a single gate with a transition matrix. Here, we instead stick to the permutation basis in the integration formula and represent a single gate as an *effective vertex* [5].
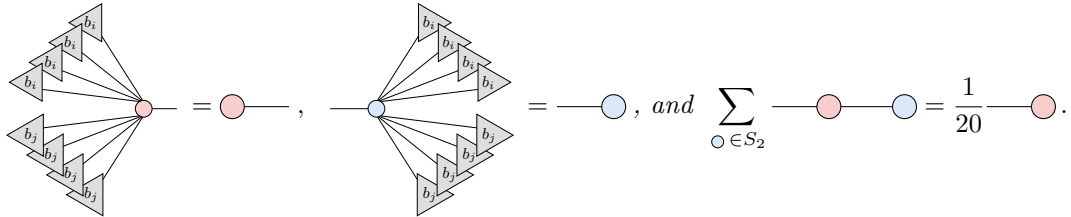
▶ **Lemma 17** ([5]). *Let U_g be a Haar-random 2-qubit gate. We have the following.*

Here $S_2 = \{\mathbb{I}, \mathbb{S}\}$ is the permutation group of two elements and an edge on the left represents four edges on the right. Specifically, the edge with σ and τ on the two ends has weight $\langle \sigma | \tau \rangle$ where

$$\langle \sigma | \tau \rangle = \begin{cases} \frac{1}{15} & , \ \sigma = \tau \\ \frac{-1}{60} & , \ \sigma \neq \tau \end{cases}.$$

for all $\sigma, \tau \in S_2$. As for the boundary condition, for each $b_i \in \{0,1\}$, we have

Apply Lemma 17 on a 1D circuit, the expectation of $\sum_{x\in\{0,1\}^n} q_C(x)^2$ is then exactly the sum over spin configurations on the the hexagonal lattice. For example, Equation 11 becomes the following. Note that each circle represents a distinct choices of elements from S_2.

$$\mathbb{E}_{C\sim\mathcal{D}}\left[\sum_{x\in\{0,1\}^n} q_C(x)^2\right] = 2^n \cdot \sum_{\bullet,\circ \in S_2} \quad \cdots \quad . \tag{12}$$

Step 2: Reducing to counting domain wall configurations on a triangular lattice

The second idea in [7] is doing local summation on the blue vertices. Specifically, he showed that the local behavior of a blue vertex and its three red neighboring vertices can be fully described only by the spin of these three red vertices.

▶ **Lemma 18** ([7, Equation 18]). *Let U_g be a Haar-random 2-qubit gate. We have the following.*

$$\sum_{\tau \in S_2} \sigma_1\!-\!\tau \begin{smallmatrix} \sigma_2 \\ \\ \sigma_3 \end{smallmatrix} = \sigma_1 \begin{smallmatrix} \sigma_2 \\ \\ \sigma_3 \end{smallmatrix} = \begin{cases} 1 & , \ \sigma_1 = \sigma_2 = \sigma_3 \\ 0 & , \ \sigma_1 \neq \sigma_2 = \sigma_3 \\ \frac{2}{5} & , \ else. \end{cases}$$

An immediate corollary of Lemma 18 is that now we can instead summing over the spin configurations over a triangular lattice. That is, Equation 12 becomes the following

$$\mathop{\mathbb{E}}_{C \sim \mathcal{D}} \left[\sum_{x \in \{0,1\}^n} q_C(x)^2 \right] = 2^n \cdot \left(\frac{1}{20} \right)^{n/2} \cdot \sum_{\bigcirc \in S_2} \quad \tag{13}$$

The advantage of working on this triangular lattice is that the non-zero term on the right hand side of Equation 13 corresponds to a *domain wall* in the triangular lattice.

▶ **Definition 19** (Domain wall). *Consider the right hand side of Equation 13 and a configuration to all the red circles. The domain wall for this configuration is a collection of disjoint horizontal lines that separate the circles that are configured to $\mathbb{I}$ from the circles that are configured to $\mathbb{S}$.*

Note that the domain wall configuration is in 1-to-1 correspondence with the spin configurations. Let dw denote a domain wall, let $w(\mathsf{dw})$ be the weight of the corresponding spin configuration. Thus, Equation 13 becomes the following.

$$\mathop{\mathbb{E}}_{C \sim \mathcal{D}} \left[\sum_{x \in \{0,1\}^n} q_C(x)^2 \right] = \left(\frac{1}{5} \right)^{n/2} \cdot \sum_{\mathsf{dw} \in} w(\mathsf{dw}) \,. \tag{14}$$

Note that a domain wall could contain two types of paths: (i) a path that goes from left boundary to the right boundary and (ii) a path that starts from and ends at both the right boundary. Furthermore, as the domain wall configuration is in 1-to-1 correspondence with subset of disjoint paths, Equation 14 becomes the following.

$$\mathop{\mathbb{E}}_{C \sim \mathcal{D}} \left[\sum_{x \in \{0,1\}^n} q_C(x)^2 \right] \leq \left(\frac{1}{5} \right)^{n/2} \cdot \left(1 + \sum_{\substack{\text{disjoint path of type (i)} \\ \in}} w(\mathsf{path}) \right) \cdot \left(1 + \sum_{\substack{\text{disjoint path of type (ii)} \\ \in}} w(\mathsf{path}) \right) \,. \tag{15}$$

Step 3: Upper bound the sum of possible path configurations of type (i)

For a set of disjoint path of type (i), it contains at most $\lfloor n/2 \rfloor$ paths. Also, a path of type (i) contributes $\left(\frac{q}{q^2+1}\right)^d$ in the weight of a domain wall.

Now, for each $1 \le t \le \lfloor n/2 \rfloor$ let us first estimate the number of domain walls having at most t paths of type (i). Specifically, for every $i \in [\lfloor n/2 \rfloor]$, the number of paths starting from i is at most 2^d because at each layer it either moves up or down. Next, for each $1 \le t \le \lfloor n/2 \rfloor$, the number of possible t paths is then at most $\binom{n/2}{t} \cdot 2^{dt}$. This gives the following upper bound for the weight contributing from domain wall of type (i).

$$
\left(\sum_{\substack{\text{disjoint path of type (i)} \\ \in \text{⬚}}} w(\text{path}) \right) \le \sum_{t=1}^{\lfloor n/2 \rfloor} \left(\frac{2}{5}\right)^{dt} \cdot \binom{\lfloor n/2 \rfloor}{t} \cdot 2^{dt}
$$

$$
\le \sum_{t=1}^{\lfloor n/2 \rfloor} \left(\frac{4}{5}\right)^{dt} \cdot \binom{\lfloor n/2 \rfloor}{t}
$$

$$
\le \left(1 + \left(\frac{4}{5}\right)^d\right)^{\lfloor n/2 \rfloor}.
$$

Consider $d \ge \frac{\log n}{\log(5/4)}$, the equation becomes

$$
\le \exp\left(\left(\frac{4}{5}\right)^{\frac{\log n}{\log(5/4)}} \frac{n}{2} \right) = O(1).
$$

Step 4: Upper bound the sum of possible path configurations of type (ii)

Let us consider the 1D circuit with infinite depth and denote the distribution as $\mathcal{D}_\infty$. Also, since the depth is infinity, the sum of the weight of domain wall with paths of type (i) is negligible. Namely, only paths of type (ii) contribute in the infinite depth circuit. Thus, we have the following upper bound.

$$
\left(1 + \sum_{\substack{\text{disjoint path of type (ii)} \\ \in \text{⬚}}} w(\text{path}) \right) \le \left(1 + \sum_{\substack{\text{disjoint path of type (ii)} \\ \in \text{⬚} \cdots}} w(\text{path}) \right)
$$

$$
= 5^{n/2} \cdot \mathop{\mathbb{E}}_{C \sim \mathcal{D}_\infty} \left[\sum_{x \in \{0,1\}^n} q_C(x)^2 \right].
$$

Finally, it is a well known fact that the expectation of the sum of squares of marginal probabilities is $O(2^{-n})$ for infinite depth 1D circuit. So the above equation becomes

$$
= O\left(\left(\frac{5}{4}\right)^{n/2} \right).
$$

Wrap up

To conclude the proof of Lemma 16, let us plug in the calculations from step 3 and step 4 into Equation 15, this gives us

$$\mathbb{E}_{C \sim \mathcal{D}}\left[\sum_{x \in \{0,1\}^n} q_C(x)^2\right] \leq \left(\frac{1}{5}\right)^{n/2} \cdot O(1) \cdot O\left(\left(\frac{5}{4}\right)^{n/2}\right) = O\left(\frac{1}{2^n}\right)$$

as desired.

References

1 Scott Aaronson and Sam Gunn. On the classical hardness of spoofing linear cross-entropy benchmarking. *arXiv preprint*, 2019. `arXiv:1910.12085`.

2 Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

3 Michael J Bremner, Ashley Montanaro, and Dan J Shepherd. Average-case complexity versus approximate simulation of commuting quantum computations. *Physical review letters*, 117(8):080501, 2016.

4 Michael J Bremner, Ashley Montanaro, and Dan J Shepherd. Achieving quantum supremacy with sparse and noisy commuting quantum computations. *Quantum*, 1:8, 2017.

5 PW Brouwer and CWJ Beenakker. Diagrammatic method of integration over the unitary group, with applications to quantum transport in mesoscopic systems. *Journal of Mathematical Physics*, 37(10):4904–4934, 1996.

6 Xun Gao and Luming Duan. Efficient classical simulation of noisy quantum computation. *arXiv preprint*, 2018. `arXiv:1810.03176`.

7 Nicholas Hunter-Jones. Unitary designs from statistical mechanics in random quantum circuits. *arXiv preprint*, 2019. `arXiv:1905.12053`.

8 John Napp, Rolando L La Placa, Alexander M Dalzell, Fernando GSL Brandao, and Aram W Harrow. Efficient classical simulation of random shallow 2d quantum circuits. *arXiv preprint*, 2019. `arXiv:2001.00021`.

9 Kyungjoo Noh, Liang Jiang, and Bill Fefferman. Efficient classical simulation of noisy random quantum circuits in one dimension. *arXiv preprint*, 2020. `arXiv:2003.13163`.

10 Guifré Vidal. Efficient simulation of one-dimensional quantum many-body systems. *Physical review letters*, 93(4):040502, 2004.

11 Man-Hong Yung and Xun Gao. Can chaotic quantum circuits maintain quantum supremacy under noise? *arXiv preprint*, 2017. `arXiv:1706.08913`.

12 Yiqing Zhou, E Miles Stoudenmire, and Xavier Waintal. What limits the simulation of quantum computers? *arXiv preprint*, 2020. `arXiv:2002.07730`.