Online Detection of Golden Circuit Cutting Points

Daniel T. Chen
Case Western Reserve University
Cleveland, OH, USA
txc461@case.edu

Ethan H. Hansen
Case Western Reserve University
Cleveland, OH, USA
ehh50@case.edu

Xinpeng Li
Case Western Reserve University
Cleveland, OH, USA
xxl1337@case.edu

Aaron Orenstein
Case Western Reserve University
Cleveland, OH, USA
aao62@case.edu

Vinooth Kulkarni
Case Western Reserve University
Cleveland, OH, USA
vxk285@case.edu

Vipin Chaudhary
Case Western Reserve University
Cleveland, OH, USA
vxc204@case.edu

Qiang Guan
Kent State University
Kent, OH, USA
qguan@kent.edu

Ji Liu
Argonne National Laboratory
Lemont, IL, USA
ji.liu@anl.gov

Yang Zhang
University of Illinois Urbana-Champaign
Champaign, IL, USA
yzhangnd@illinois.edu

Shuai Xu
Case Western Reserve University
Cleveland, OH, USA
sxx214@case.edu

Abstract—Quantum circuit cutting has emerged as a promising method for simulating large quantum circuits using a collection of small quantum machines. Running low-qubit circuit "fragments" not only overcomes the size limitation of near-term hardware, but it also increases the fidelity of the simulation. However, reconstructing measurement statistics requires computational resources—both classical and quantum—that grow exponentially with the number of cuts. In this manuscript, we introduce the concept of a golden cutting point, which identifies unnecessary basis components during reconstruction and avoids related downstream computation. We propose a hypothesis-testing scheme for identifying golden cutting points, and provide robustness results in the case of the test failing with low probability. Lastly, we demonstrate the applicability of our method on Qiskit's Aer simulator and observe a reduced wall time from identifying and avoiding obsolete measurements.

Index Terms—quantum circuit cutting, circuit cutting, circuit knitting, circuit reconstruction, hypothesis-testing, golden cutting point

I. INTRODUCTION

Quantum circuit cutting refers to the method of splitting quantum circuits into a set of small independent circuit fragments [1]. Using circuit cutting methods, large quantum circuits can be simulated by a collection of smaller machines, barring some addition classical computing resources. Moreover, it was also empirically shown that cutting the circuit reduces the affect of noise [2], [3] and can be used for error mitigation [4]. There has also been work on properly accounting for statistical shot noise [5] and adaptation to specific problems such as combinatorial optimization [6]. Thus, this technique holds great promise for resolving many practical issues with utilizing quantum hardware, particularly in the NISQ era [7].

However, circuit cutting suffers greatly in practice as the runtime grows exponentially with the number of cuts. Akin to

quantum state tomography, circuit cutting works by classically tracking all quantum degrees of freedom at the cut locations. Thus, exponential scaling comes naturally as the quantum state of interest grows. There have been many efforts to reduce this cost through randomized measurements [8], [9], classical sampling [10], and variational optimization [11]. Nonetheless, the exponential growth in runtime is unlikely to vanish without imposing structural assumptions on the circuit. Alternatively, finding applications of circuit cutting that avoid the scaling issue, as demonstrated in [4], also remains a problem of interest.

In this manuscript, we build upon our previous work [12] and propose an algorithm for online detection of neglectable basis elements during circuit cutting. In [12], we showed that some reconstruction procedures can be sped up if we impose extra assumptions on the circuit—namely, whether a basis element can be neglected. However, such assumptions cannot be easily detected *a priori*. Thus, we propose a hypothesis testing scheme at each cut location that, at no additional cost in run time, identifies whether there is statistically significant evidence against the assumption being true. We empirically demonstrate the viability of our method on the Qiskit simulator [13], and examine scaling effects with respect to important algorithm parameters.

The outline of the paper is as follows. Section II re-derives the circuit cutting in the general bipartition case and introduces the concept of a "golden cut"—a cut location that has basis elements which can be neglected. We also show the algorithm for detecting golden cuts as well as their statistical properties. In Section III, we demonstrate the applicability of our method in a simple, one-cut case. Meanwhile, we explore additional properties of the proposed algorithm under varying parameters. Lastly, we discuss some future directions in Section IV.
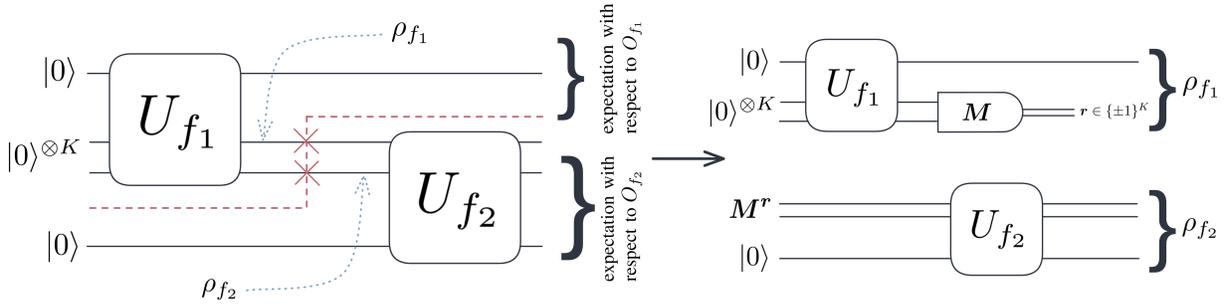
Fig. 1: An example of circuit bipartition using K cuts. The circuit is comprised of arbitrary unitary gates U_{f_1} and U_{f_2} such that the middle K qubits are possibly entangled. The circuit can be split into two fragments by performing K cuts between the two gates, and classically recombining measurement outcomes in basis M and preparing eigenstates of M for a collection of operators that forms a basis over density matrices.

II. THEORY

We start the section by briefly introducing circuit cutting only for the bipartite case. Readers are encouraged to consult [1], [5], [9] for more detailed derivations and/or general treatment of circuit cutting. Section II-A ends with the definition of golden cuts—circuit structure that induces neglectable basis elements. Then, Section II-B derives an online detection algorithm for these structures.

A. Circuit Bipartition

Suppose an N-qubit quantum circuit induces a state ρ. Further suppose we want to perform K cuts on a quantum circuit to divide it into two fragments, f_1 and f_2 (see Figure 1). Indexing each cut by an integer in $[K]$, then we can write the cutting scheme as an injective function $c : [K] \to [N]$ that maps a cut to the respective qubit being cut. The premise of circuit cutting is to rewrite ρ in terms of the states induced by the circuit fragments, ρ_{f_1} and ρ_{f_2}, albeit with some parameterization.

As first shown in Peng *et al.* [1], this can be done via expanding ρ at the location of the cut. That is, for a basis set over 2×2 Hermitian matrices, $\mathcal{B} = \{I, X, Y, Z\}$, the following decomposition holds:

$$\rho = \frac{1}{2^K} \sum_{M \in \mathcal{B}^K} \rho_{f_1}(M) \otimes \rho_{f_2}(M) \quad (1)$$

where

$$M = \begin{pmatrix} M_{c(1)} & M_{c(2)} & \dots & M_{c(K)} \end{pmatrix}, \quad M_{c(i)} \in \mathcal{B}. \quad (2)$$

The state of each fragment ρ_{f_i}, $i = 1, 2, \dots$ is parameterized by an operator M and depends on the particular gates contained in the circuit. Let U_{f_i} denote the unitary operation induced by the quantum gates on each fragment, and let $|0\rangle$

be an N_{f_1}-qubit "zero" state. Then, we can write ρ_{f_i} as the following (up to appropriate qubit permutation):

$$\rho_{f_1}(M) = \mathrm{tr}_{c(1), \dots, c(K)} \left(\bigotimes_{i \in [K]} M_{c(i)} U_{f_1} |0\rangle \langle 0| U_{f_1}^\dagger \right), \quad (3)$$

$$\rho_{f_2}(M) = U_{f_2} \left(\bigotimes_{i \in [K]} M_{c(i)} \otimes |0\rangle \langle 0| \right) U_{f_2}^\dagger. \quad (4)$$

The choice of basis is arbitrary, and we chose the normalized Pauli basis for simplicity. Note that the above equation lacks a physical interpretation as elements in $\mathcal{B}$ are traceless (except for I) and hence, are not quantum states.

To resolve this issue, we note that each operator M admits spectral decomposition. Letting

$$r = \begin{pmatrix} r_{c(1)} & r_{c(2)} & \dots & r_{c(K)} \end{pmatrix} \in \{\pm 1\}^K \quad (5)$$

be a tuple of eigenvalues, we define

$$M^r = \begin{pmatrix} M_{c(1)}^{r(1)} & M_{c(2)}^{r(2)} & \dots & M_{c(K)}^{r(K)} \end{pmatrix} \quad (6)$$

to be the r-th eigenstate of operator M. Let $s \in \{\pm 1\}^K$ and M^s be similarly defined. Applying this decomposition gives the reconstruction formula in the bipartition case:

$$\rho = \frac{1}{2^K} \sum_{\substack{M \in \mathcal{B}^K, \\ r, s \in \{\pm 1\}^K}} \mathrm{Par}(r) \, \mathrm{Par}(s) \, \rho_{f_1}(M^r) \otimes \rho_{f_2}(M^s) \quad (7)$$

where $\mathrm{Par}(r)$ denotes the parity of a string of eigenvalues, i.e., $\mathrm{Par}(r) = \prod_i r_i$. The formula above lends itself to a measure-and-prepare scheme for realizing quantum circuit cutting: for each basis element M, we measure the upstream circuit in the basis, prepare the downstream circuit into the eigenstates of the same basis, and reweight the outcome of the downstream circuit by the probability of observing the respective eigenstate upon measuring the upstream fragment.

Alternatively, for any desired quantum observable O, suppose the operator can be decomposed to accommodate the two fragments, i.e., $O = O_{f_1} \otimes O_{f_2}$ up to appropriate permutation of qubit indices. Then, we can arrive at an expression for the

expectation of the uncut circuit in terms of the fragments ρ_{f_i} and their respective observables O_{f_i}:

$$
\begin{aligned}
&\text{tr}(O\rho) \\
&= \frac{1}{2^K} \sum_{M, r, s} \text{Par}(r)\text{Par}(s)\text{tr}\left((O_{f_1} \otimes O_{f_2})(\rho_{f_1} \otimes \rho_{f_2})\right) \quad (8) \\
&= \frac{1}{2^K} \sum_{M, r, s} \text{Par}(r)\text{Par}(s)\,\text{tr}\left(O_{f_1}\rho_{f_1}\right)\text{tr}\left(O_{f_2}\rho_{f_2}\right) \quad (9)
\end{aligned}
$$

where we implicitly apply M to ρ_{f_1} and ρ_{f_2}.

Note that the decomposition assumption $O = O_{f_1} \otimes O_{f_2}$ is without loss of generality. For any choice of Hermitian operator O, one can expand it with respect to Pauli strings, i.e.,

$$
O = \sum_{S \in \mathcal{B}^N} a_S\, S \tag{10}
$$

for some set of real coefficients $\{a_S\}$. So, by linearity of the trace operator, we obtain a generalized expression for the expectation:

$$
\begin{aligned}
&\text{tr}(O\rho) \\
&= \frac{1}{2^K} \sum_{S, M, r, s} a_S \text{Par}(r)\text{Par}(s)\text{tr}\left(S_{f_1}\rho_{f_1}\right)\text{tr}\left(S_{f_2}\rho_{f_2}\right) \quad (11)
\end{aligned}
$$

where S_{f_1} and S_{f_2} are the Pauli strings separated according to the circuit cutting scheme, i.e., $S = S_{f_1} \otimes S_{f_2}$ under appropriate qubit permutations.

We now formally define the *golden circuit cutting point*.

Definition 1. *Consider an N-qubit circuit amenable to bipartition with K cuts. We're interested in the expectation of the circuit-induced state with respect to some quantum observable $O = O_{f_1} \otimes O_{f_2}$. The cutting scheme admits a* golden cutting point *if there exists $M_* \in \mathcal{B}^K$ such that*

$$
\sum_{r \in \{\pm 1\}^K} \text{Par}(r)\,\text{tr}\left(O_{f_1}\rho_{f_1}(M_*^r)\right) = 0 \tag{12}
$$

More simply put, a golden cutting point refers to the existence of a basis element that leads to systematic cancellations. Golden cutting points neither necessarily exists nor are unique. For each such basis element, one does not need to execute the circuit downstream of the cut with initialization corresponding to the neglected basis.

Golden cutting points can be constructed via circuit design by restricting the set of rotations allowed prior to cutting so long as the structure of the quantum circuit permits. However, one should not expect such a property to hold for an arbitrary algorithm. Thus, we propose an "online" scheme for detecting the existence of golden cutting points and establish robustness of misidentifying golden cuts.

B. Identifying Golden Cutting Points

With no knowledge of the existence of golden cutting points, one must execute each of the 4^K upstream circuits and another 4^K downstream circuits (f_1 and f_2 respectively in the bipartition case). To detect golden cutting points in the absence of *a priori* knowledge, we propose to conduct a hypothesis test for each of the 4^K upstream circuits, determine whether there is statistically significant evidence for the existence of a golden cutting point, then run the corresponding downstream circuit.

Denote

$$
\tau = \sum_{r \in \{\pm 1\}^K} \text{Par}(r)\,\text{tr}\left(O_{f_1}\rho_{f_1}(M_*^r)\right) \tag{13}
$$

as the quantity we want to verify magnitude of. Inheriting the bipartition assumption from the previous section, we can rewrite τ as estimating the expectation of a larger observable

$$
\tau = \text{tr}\left((O_{f_1} \otimes M_*)\,U_{f_1}|0\rangle\langle0|U_{f_1}^\dagger\right). \tag{14}
$$

where, again, $U_{f_1}|0\rangle$ is the state induced by the upstream fragment. Writing τ in this form allows us to employ standard techniques for estimating quantum observables.

Assume for convenience that O_{f_1} is a Pauli-string. To estimate the expectation, we measure each qubit in the respective Pauli basis (by performing a rotation V) m times and obtain an ensemble of bitstring samples $\{\hat{b}_i\}_{i=1}^m$. Therefore, we can estimate τ be constructing

$$
\hat{\tau} = \frac{1}{m} \sum_{i=1}^m \langle\hat{b}_i|V^\dagger(O_{f_1} \otimes M)V|\hat{b}_i\rangle. \tag{15}
$$

Alternatively, one can think of estimating the distribution of strings of eigenvalues (which we'll call eigenstrings for short) induced by the measurements. Write p_b for the probability of obtaining eigenstring b, and $\hat{p}_b$ for the empirical probability. Moreover, let p and $\hat{p}$ denote the vector of probabilities. Then, we can sum the parity of each eigenstring weighted by the (empirical) probability to arrive at an alternative expression for the estimator:

$$
\hat{\tau} = \sum_{b \in \{\pm 1\}^{N_{f_1}}} \text{Par}(b)\,\hat{p}_b. \tag{16}
$$

The proposition below establishes the standard error and asymptotic normality, which are convenient for hypothesis testing.

Proposition 1 (Asymptotic Normality)**.** *Given a circuit amenable to the bipartite circuit cutting scheme (cf. II-A), let $\hat{\tau}$ be the estimator of τ expressed in Equation 15 and let O_{f_1} admit decomposition as in Equation 10. Then, $\hat{\tau}$ is asymptotically normal, i.e.,*

$$
\frac{\hat{\tau} - \tau}{\text{std}(\hat{\tau})} \to \mathcal{N}(0, 1) \tag{17}
$$

where the standard deviation of the estimator is expressed as

$$
\text{std}(\hat{\tau}) = \left(\sum_{S \in \mathcal{B}^{N_{f_1}}} \frac{a_S^2}{N} \chi^\intercal(\text{diag}(\hat{p}_S) - \hat{p}_S\hat{p}_S^\intercal)\chi\right)^{1/2} \tag{18}
$$

and χ is the vector of parities, i.e., $\chi_b = \text{Par}(b)$.

Proof. We first consider the case were O_{f_1} is a Pauli string S, then proceed to generalize to arbitrary quantum observables.

Estimating the expectation of the Pauli string $\boldsymbol{S}$ with m shots, using the formalism presented in Equation 16, gives the standard error

$$\text{Var}(\hat{\tau}) = \text{Cov}\left(\sum_b \hat{p}_b \text{ Par}(b), \sum_{b'} \hat{p}_{b'} \text{ Par}(b')\right) \quad (19)$$

$$= \sum_{b,b'} \text{Cov}(\hat{p}_b, \hat{p}_{b'}) \text{Par}(b)\text{Par}(b') \quad (20)$$

$$= \frac{1}{m}\left(\sum_{b=b'} p_b(1-p_b) - \sum_{b \neq b'} p_b p_{b'} \text{Par}(b)\text{Par}(b')\right) \quad (21)$$

where the third equality follows from the covariance of multi-nomial distributions. Using the empirical quantities for each p_b and writing in matrix form gives the estimated standard error

$$\text{std}(\hat{\tau}) = \sqrt{\frac{1}{m}\boldsymbol{\chi}^{\mathsf{T}}(\text{diag}(\hat{\boldsymbol{p}}_S) - \hat{\boldsymbol{p}}_S\hat{\boldsymbol{p}}_S^{\mathsf{T}})\boldsymbol{\chi}}. \quad (22)$$

Consider the decomposition in Equation 10. As the estimation procedure runs independently, variances add. Hence, we arrive at the form in Equation 18.

Lastly, asymptotic normality is established by the equivalence formulation presented in Equation 15 and Equation 16. In the form of Equation 15, we can express $\hat{\tau}$ as the average over independent samples. In combination with finiteness of O_{f_1}, the Central Limit Theorem holds, implying asymptotic normality of $\hat{\tau}$. □

Using the above proposition, we can deduce an algorithm for detecting golden cutting points. For each basis element in $\mathcal{B}^K$, we will compute $\hat{\tau}$ and perform a statistical test for whether $\tau \neq 0$. If we've gathered statistically significant evidence for τ being non-zero, then we would run the downstream fragment parameterized by the respective basis element. On the other hand, if $\hat{\tau}$ is sufficiently close to zero, then we classify the cut as a golden cutting point and proceed without running the corresponding downstream fragment. Using Φ to denote the CDF of a standard Gaussian, we summarize the above procedure in Algorithm 1.

While we can control the rate of correctly identifying golden cutting points by the significance level α, we would also like to derive a way of controlling the rate of false negatives. In fact, falsely identifying a non-golden cutting point as golden is more problematic than falsely identifying golden as non-golden. This is because in the latter case the reconstructed bitstring distribution will likely still be within acceptance error ranges, but that's not the case for the former. Thus, we hope to lower the probability of false negatives by taking a sufficient number of shots during the estimation procedure.

Remark 1. *As we have showed that the estimator converges weakly to a normal distribution, we will assume Gaussianity to facilitate analysis. Suppose $\hat{\tau} \sim \mathcal{N}(\tau, b^2/m)$ where $b = \sqrt{m}\cdot\text{std}(\hat{\tau})$. We hope that for a basis element where $\tau > \epsilon > 0$, the hypothesis testing scheme would reject it, perhaps with a*

Algorithm 1: Online detection of golden cutting points

Input: fragments f_1 and f_2, observable O_{f_1}, O_{f_2}, significance level $\alpha \in (0, 1)$
Output: Expectation $\text{tr}(O\rho)$

1 **for** $M \in \mathcal{B}^K$ **do**
2   Compute $\hat{\tau}$ using Eqn. 16
3   **if** $|\hat{\tau}| > \Phi^{-1}(1-\alpha) \cdot \text{std}(\hat{\tau})$ **then**
4   Reject the hypothesis and compute $\text{Par}(\boldsymbol{s}) \text{ tr}(O_{f_2}\rho_{f_2}(\boldsymbol{M}^{\boldsymbol{s}})$ for all $s \in \{-1, +1\}^K$
5   **else**
6   Fail to reject the hypothesis and set quantities related to $\boldsymbol{M}$ to zero
7 Reconstruct the full expectation from fragment data using Eqn. 9

---

small probability of error δ. By the Chernoff bound, we know that

$$\Pr(|\hat{\tau} - \tau| > \epsilon) \leq 2e^{-m\epsilon^2/2b^2} = \delta. \quad (23)$$

Thus, to estimate τ to any desired ϵ accuracy with probability $1 - \delta$, we need

$$m \geq \frac{2b^2}{\epsilon^2}\log\frac{2}{\delta} \quad (24)$$

measurements. Since b is not known a priori, we can upper bound it by

$$\mathbf{1}^{\mathsf{T}}(\text{diag}(\boldsymbol{q}) + \boldsymbol{q}\boldsymbol{q}^{\mathsf{T}})\mathbf{1}. \quad (25)$$

The distribution $\boldsymbol{q}$ that maximizes the above quantity is the uniform distribution. Thus, we can arrive at a definitive upper bound $b \leq \frac{3}{2}(1 - 2^{-N_{f_1}})$.

Alternatively, one can interpret the proposed sample complexity as accepting an ϵ margin for Equation 12 in the sense that we declare a cut is golden if $|\hat{\tau}| < \epsilon$, thereby accepting an additional additive error of magnitude $\mathcal{O}(\epsilon)$ to the result of the circuit reconstruction. On the other hand, if $\tau > \epsilon$, we wish to identify it with probability $1 - \delta$. Note that in the limit of $m \to \infty$, variance vanishes and the true positive and negative rates approach one.

III. Experiments

In this section, we numerically demonstrate the applicability of Algorithm 1. The algorithm was implemented in Qiskit and executed on the Aer simulator. We examine its statistical (Section III-A) and runtime (Section III-B) properties through studying its dependency on the number of shots and the significance level α.

A. Statistical Analysis

As is standard in analyzing binary decisions, we analyzed our statistical test by providing instances when the null hypothesis is true and when it's false. Specifically, we provided circuits either with or without golden cuts, and observed the
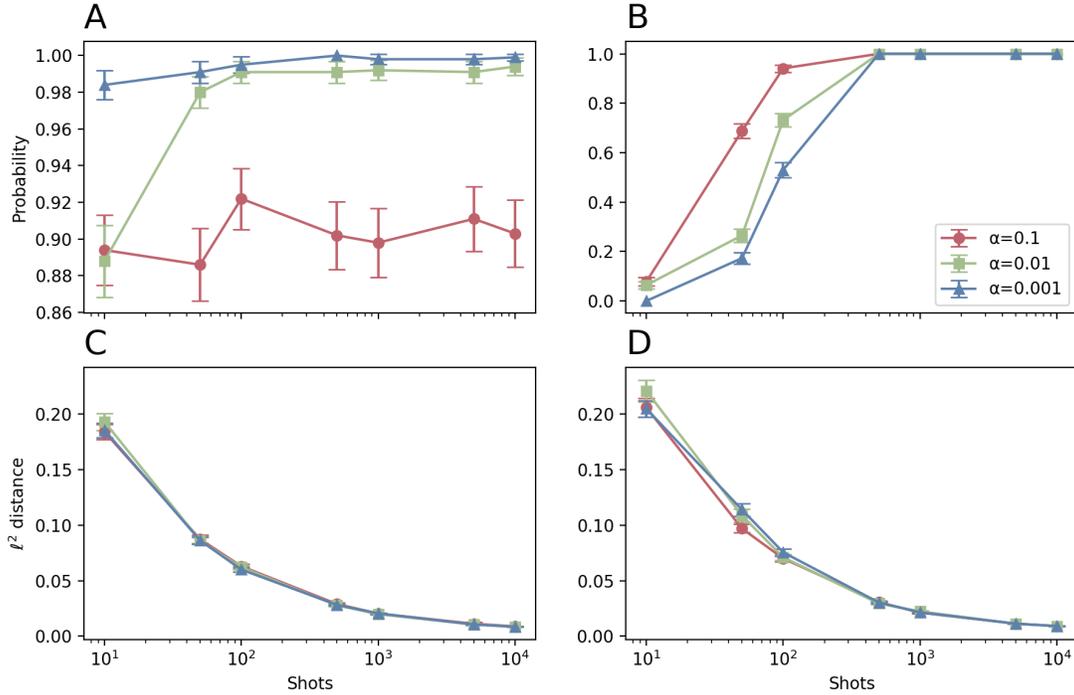
Fig. 2: Behavior of Algorithm 1 for varied shots and α levels averaged over 1000 independent trials. The rate of true positives (**A**) is consistent with the specified α, and the true negative rate (**B**) converges to one as the number of shots increases. The reconstruction error monotonically vanishes both in the presence (**C**) and the absence (**D**) of golden cuts.

probability of correctly identifying the existence (or lack) of golden cuts.

We consider a simple three-qubit circuit amenable to cutting on the second qubit. First, we generated a circuit containing a golden cut by appending two R_X gates on the first and second qubits, then an R_Y gate only on the first qubit—this is U_{f_1} from Figure 1, and we let $K = 1$. Rotation angle θ was set to a value far from zero ($\theta = 0.5$) to ensure X would be the only golden cutting axis and to focus only on statistical shot noise. To generate a circuit known to not contain a golden cutting point, we applied the same procedure then appended an additional R_Y gate on the second qubit (the qubit being cut) as well. Finally, we generated U_{f_2} randomly across qubits 2 and 3 using Qiskit's `random_circuit` function. Once the circuit was constructed, for each shot count-α pair, we repeated 1000 independent executions of Algorithm 1 and collected the frequency at which the algorithm correctly identified the circuit structure. Results are displayed in subplots A and B of Figure 2.

Subplot A shows the probability of failing to reject the null hypothesis given the null hypothesis is true, which should be exactly α. The numerics aligned with the theoretical value with the exception of cases with low shot counts. This can be understood as our estimator is built upon asymptotic statements on the sampling distribution. Subplot B demonstrates the rate of true negatives. We can see that, given sufficient samples, we always identified non-golden cuts correctly. For lower significance values, we are more prone to rejecting

the null hypothesis, explaining the faster rate of convergence towards 1 for smaller α-values.

We also examined the quality of the reconstruction by calculating the distance between the empirical and theoretical bitstring distributions. The theoretical distribution is obtained by taking large number of shots without circuit cutting. We employed the ℓ^2-distance to quantify how far apart two distributions are, i.e., for discrete distributions $\boldsymbol{p}$ and $\boldsymbol{q}$,

$$d(\boldsymbol{p}, \boldsymbol{q}) = \sqrt{\sum_i (p_i - q_i)^2}. \quad (26)$$

Again, we executed Algorithm 1 independently for 1000 trials and collected the ℓ^2 distance between the empirical, reconstructed bitstring distribution and the respective theoretical distribution at varying numbers of circuit execution shots and alpha levels. Results are found in subplots C and D of Figure 2.

In general, the reconstruction error decreases monotonically with the number of shots, and there was not a significant difference among choices of α. In the case of low shot count and no existing golden cut, there seems to be more statistical fluctuation when reconstructing. Considering subplot B above, we know that this region is prone to false negatives, and thus neglecting bases that should not be neglected.

B. Runtime Analysis

To obtain timed runtime values, we generated a circuit with a golden cut and executed Algorithm 1. Recall that, depending

		run time (sec)	
		w/ optimization	w/o optimization
α	10^{-1}	0.0771 ± 0.0006	0.0959 ± 0.0004
	10^{-2}	0.0749 ± 0.0004	0.0961 ± 0.0004
	10^{-3}	0.0747 ± 0.0003	0.0962 ± 0.0004

TABLE I: Runtime comparison between circuit cutting procedures with and without optimization from Algorithm 1. Independent trials were repeated 1000 times. We can see that neglecting basis elements consistently run faster despite spending computing overhead on hypothesis testing.

on the results of the hypothesis test on the upstream circuit, the downstream circuit might not be executed for certain bases. Then, we ran the same cut circuit and performed the usual reconstruction routine without hypothesis testing or golden cutting optimization. Both of these processes were timed individually over 1000 trials and at varying alpha levels. Results for this can be found in Table I. In general, we see roughly a 20% decrease in runtime upon performing the optimization. As α decreases, we tend to reject the null hypothesis more often, thereby executing the downstream circuit more often.

IV. CONCLUSION

In this manuscript, we proposed an online detection algorithm for finding golden cutting points—circuit structures that induce neglectable basis elements during circuit reconstruction. The detection algorithm was built on performing a hypothesis test for each basis element, and executing the respective downstream circuit only if the null hypothesis is rejected. The detection does not require additional circuit executions. We showed numerically that, under sufficient number of shots, golden cuts will always be detected and there is no drastic difference among choices of significance levels.

Empirically testing the algorithm on quantum hardware and at large scale—both of which contribute additional noise that can affect the quality of estimators—are left to future work. Another immediate open question is the prevalence of circuit structure amenable to golden cuts in applicable circuits. For instance, we found that the SupermarQ [14] and QASMBench [15] benchmark suites both feature a handful of benchmarks that exhibit this circuit structure. Variational circuits whose ansatz can be flexible might also be a candidate to apply golden cutting point restrictions for better scalability.

REFERENCES

[1] T. Peng, A. W. Harrow, M. Ozols, and X. Wu, "Simulating large quantum circuits on a small quantum computer," *Physical Review Letters*, vol. 125, no. 15, oct 2020. [Online]. Available: https://doi.org/10.1103/PhysRevLett.125.150504

[2] T. Ayral, F.-M. Le Régent, Z. Saleem, Y. Alexeev, and M. Suchara, "Quantum divide and compute: Hardware demonstrations and noisy simulations," in *2020 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*. IEEE, 2020, pp. 138–140. [Online]. Available: https://ieeexplore.ieee.org/document/9155024

[3] T. Ayral, F.-M. L. Régent, Z. Saleem, Y. Alexeev, and M. Suchara, "Quantum divide and compute: exploring the effect of different noise sources," *SN Computer Science*, vol. 2, no. 3, pp. 1–14, 2021. [Online]. Available: https://doi.org/10.1007/s42979-021-00508-9

[4] J. Liu, A. Gonzales, and Z. H. Saleem, "Classical simulators as quantum error mitigators via circuit cutting," *arXiv preprint arXiv:2212.07335*, 2022. [Online]. Available: https://arxiv.org/abs/2212.07335

[5] M. A. Perlin, Z. H. Saleem, M. Suchara, and J. C. Osborn, "Quantum circuit cutting with maximum-likelihood tomography," *npj Quantum Information*, vol. 7, no. 1, p. 64, 2021. [Online]. Available: https://doi.org/10.1038/s41534-021-00390-6

[6] Z. H. Saleem, T. Tomesh, M. A. Perlin, P. Gokhale, and M. Suchara, "Quantum Divide and Conquer for Combinatorial Optimization and Distributed Computing," *arXiv preprint*, Jul. 2021. [Online]. Available: https://arxiv.org/abs/2107.07532

[7] J. Preskill, "Quantum computing in the nisq era and beyond," *Quantum*, vol. 2, p. 79, 2018.

[8] A. Lowe, M. Medvidović, A. Hayes, L. J. O'Riordan, T. R. Bromley, J. M. Arrazola, and N. Killoran, "Fast quantum circuit cutting with randomized measurements," 2022. [Online]. Available: https://arxiv.org/abs/2207.14734

[9] D. T. Chen, Z. H. Saleem, and M. A. Perlin, "Quantum divide and conquer for classical shadows," *arXiv preprint arXiv:2212.00761*, 2022. [Online]. Available: https://arxiv.org/abs/2212.00761

[10] D. Chen, B. Baheri, V. Chaudhary, Q. Guan, N. Xie, and S. Xu, "Approximate quantum circuit reconstruction," in *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 2022, pp. 509–515.

[11] G. Uchehara, T. M. Aamodt, and O. Di Matteo, "Rotation-inspired circuit cut optimization," *arXiv preprint arXiv:2211.07358*, 2022. [Online]. Available: https://arxiv.org/abs/2211.07358

[12] D. T. Chen, E. H. Hansen, X. Li, V. Kulkarni, V. Chaudhary, B. Ren, Q. Guan, S. Kuppannagari, J. Liu, and S. Xu, "Efficient quantum circuit cutting by neglecting basis elements," *arXiv preprint arXiv:2304.04093*, 2023.

[13] A. tA-v *et al.*, "Qiskit: An open-source framework for quantum computing," 2021.

[14] T. Tomesh, P. Gokhale, V. Omole, G. S. Ravi, K. N. Smith, J. Viszlai, X.-C. Wu, N. Hardavellas, M. R. Martonosi, and F. T. Chong, "Supermarq: A scalable quantum benchmark suite," in *2022 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, 2022, pp. 587–603.

[15] A. Li, S. Stein, S. Krishnamoorthy, and J. Ang, "Qasmbench: A low-level quantum benchmark suite for nisq evaluation and simulation," *ACM Transactions on Quantum Computing*, vol. 4, no. 2, feb 2023. [Online]. Available: https://doi.org/10.1145/3550488

[16] S. Greb, "Nord theme." [Online]. Available: https://www.nordtheme.com/