SCALABLE BENCHMARKING OF QUANTUM CHEMISTRY ALGORITHMS USING CIRCUIT MIRRORING

AIDAN Q. WILBER-GAUTHIER* AND STEFAN K. SERITAN†

Abstract. Current benchmarking methods for measuring progress towards applications on quantum computers often lack scalability and specificity. This limits their ability to produce generalizable metrics of processor performance. Here, we describe a method for creating scalable application-specific benchmarks using mirrored subcircuits that quantify a device's ability to execute particular subroutines or entire algorithms. We then apply this method to two quantum chemistry subroutines, performing noisy numerical simulations and interpreting their results within the paradigm of volumetric benchmarking. We show how to extract effective error rates and predict full circuit fidelities from the subcircuit data and demonstrate that we can distinguish differences in performance resulting from structural properties of the circuits.

1. Introduction. Quantum processors promise to revolutionize computing, possessing the ability to solve problems that are intractable on classical computers. Significant developments have been made in creating algorithms that may soon realize useful applications of quantum computing. Yet, the most interesting applications of quantum devices are stalled by complex and hard-to-predict hardware errors present in them. Many protocols have been developed to characterize such errors and evaluate the performance of quantum processors. Tomographic approaches such as gate set tomography (GST) [3] produce detailed models of device noise, but are currently not scalable beyond a few qubits. Other methods, such as randomized benchmarking (RB) [19] or IBM's quantum volume benchmark [5], scale to at least a reasonable number of qubits, but provide high-level metrics of device performance that are not indicative of how well a device can execute real-world tasks. Novel benchmarking methods that reconcile scalability with the ability to provide tailored performance metrics are needed to better measure the capability of quantum devices.

As modern quantum processors approach the size necessary for small but potentially useful applications, there is growing interest in application-specific benchmarks that probe a device's performance on a particular algorithm. Most current methods choose a set of one or more tasks to create a benchmarking suite, aiming to capture especially common or relevant programmatic structures that reflect realistic computations [1, 6, 7, 9, 12, 13, 14, 15]. Other recent works have presented frameworks for creating application-specific benchmarks for arbitrary applications, known as benchmark generators [8, 11]. Benchmarks created using these methods rely on either exponentially scaling classical simulation of the benchmarking circuits, exploitation of some problem-specific property that allows one to construct easily predictable circuits, or some other application-specific strategy for quantifying the quality of the benchmarking circuit outcomes. Scalable benchmarks can only be created using these frameworks if one implements a clever circuit construction or outcome verification strategy, detracting from the generality these methods aim to provide.

Here, we introduce a method for creating scalable volumetric benchmarks for almost any algorithm. From an application circuit, we sample a set of subcircuits of varying widths and depths that is representative of the full circuit. We then estimate the subcircuit fidelities using mirror circuit fidelity estimation (MCFE) [18], which provides an efficient, generalizable method for characterizing the performance of the subcircuits, resulting in application-agnostic scalability. We apply this method to two quantum chemistry application circuits and show how subcircuit fidelities can be used as a proxy for full circuit performance using effective error rates. After establishing the efficacy of these error rates, we demonstrate

*Sandia National Laboratories, aqwilbe@sandia.gov
†Sandia National Laboratories, sserita@sandia.gov

that our benchmark is able to reveal performance differences between the two algorithms resulting from structural properties of the circuits.

2. Benchmark Generation. In this section we describe our method for generating application-specific benchmarks. Starting from an application-circuit $\tilde{C}$, we first pass the circuit through a minimal compilation procedure, which maps the circuit to our target device's qubit graph containing only one- and two-qubit gates. The mapped circuit, written as a composition of w_C-qubit layers $C = L_{d_C} \cdots L_1$, is passed to one of our subcircuit selection routines, described in Section 2.1, producing a set $\{S_C\}$ of subcircuits with varying, user-defined shapes. The subcircuits are then mirrored resulting in a set of mirror circuits $\{M_C\}$ to run on the target device, after which we use mirror circuit fidelity estimation (MCFE) to quantify the performance of the subcircuits, as described in Section 2.2.

2.1. Subcircuit Selection. The goal of subcircuit selection is to generate a set of subcircuits that are representative of the full circuit. The primary difficulty in this task is the presence of two-qubit gates, since one cannot simply choose any range of circuit layers and subset of qubits without producing dangling gates — two-qubit gates where one qubit is in the selected subset and the other is not. In our simple method (Section 2.1.1), we choose to drop dangling gates from the subcircuit, whereas our connected components method (Section 2.1.2) avoids dangling gates entirely. In both cases, the user defines a set of subcircuit shapes (i.e., width-depth pairs) and chooses how many subcircuits to sample for each shape.

2.1.1. Simple Method. In our simple selection method, we pick subcircuits iteratively over all the width-depth pairs until the desired number of subcircuits is reached. To pick a subcircuit of width w and depth[1] d, we first generate the set of all connected subgraphs of w qubits on which the circuit is mapped. A starting layer L_i is chosen uniformly from the set of allowable starting layers $\{L_1, \ldots, L_{d_C - d}\}$ along with a subset of w connected qubits. Layers are added until the target depth is reached, dropping and counting dangling gates. While this method works well for small full circuit and subcircuit sizes, the generation of the connected qubit subgraphs is combinatorially difficult, limiting its use for circuits using more than a few dozen qubits. The restriction of uniformly sampling connected qubit subgraphs can be lifted to remove this computational bottleneck, but this was not explored in the current work.

2.1.2. Connected Components Method. In the connected components method, we consider each possible starting layer L_i, and partition the qubits into disjoint subsets that are not connected by a two-qubit gate. From the starting layer, we iterate over subsequent layers, unioning subsets when a two-qubit gate is found that connects them. Concurrently, we track the depth of each subset, ensuring that "inactive" layers (i.e., layers without operations on the chosen subset) are not counted. Whenever a w-qubit subset with depth d where (w, d) is a desired shape is found, we record the layers and qubits as a potential subcircuit, since we might find more subcircuits than desired. After our search is finished, we choose a random subset of the subcircuit options from each shape as our final set of subcircuits. In contrast to the simple method, this process is computationally efficient, but it does not always result in a sufficient number of subcircuits. Optionally, we can sample wider subcircuits by combining parallel subcircuits on disjoint sets of qubits or deeper subcircuits by dropping qubits from high-depth, high-width subcicuits, but even these steps do not guarantee the necessary number of subcircuits.

[1]For our purposes, we take depth to mean the *physical* depth, or how many unit-time 'cycles' the circuit takes to execute on the target device.

2.2. Mirror Circuit Fidelity Estimation. Here, we provide a brief overview of circuit mirroring [17], which underpins both mirror randomized benchmarking (RB) [19], a variant of RB that can scale to hundreds of qubits, as well as the mirror circuit fidelity estimation (MCFE) [18] technique that we will use to probe the performance of each subcircuit. Circuit mirroring is based on the simple idea that an error-free quantum circuit should be perfectly reversible, i.e. running the circuit forwards and backwards should return the system back to the initial state. Therefore, any deviation from the initial state must be due to errors in the circuit. If the initial state is known or easy to prepare, then one can directly measure this deviation and thus how noisy the circuit execution was. However, running the "mirrored" (i.e. combined forward and backward) circuit by itself is known as the Loschmidt echo, and is not sufficient to capture all the errors in the system. Thus, the version of circuit mirroring in MCFE uses three additional features: i) randomized compiling to twirl coherent errors and ensure they do not systematically cancel, ii) using ensembles of local random initial states with classical postprocessing to avoid needing complicated n-qubit initial states, and iii) reference experiments to correct for state preparation and measurement (SPAM) errors.

Calculating the fidelity of subcircuits with MCFE has several advantages that make it useful for creating application-inspired benchmarks. Firstly, it captures both stochastic and coherent errors. This is particularly important in the case of algorithmic circuits, which have more structure than the random circuits used in many other benchmarks and may interact with coherent errors in nontrivial ways. Secondly, it is a scalable technique due to the localized nature of both the random states and the randomized compilation, making the classical processing of the circuits to only scale linearly with number of circuit locations. This allows for the fidelity of even large subcircuits to be estimated efficiently using MCFE.

3. Quantum Chemistry Applications. Quantum chemistry has been identified as one possible "killer app" for quantum computing [20], and as such there has been much effort in developing quantum algorithms for chemistry and materials science in recent years. One common use case is ground state energy estimation, where the ground state of some molecular Hamiltonian is prepared on the quantum computer and quantum phase estimation (QPE) is used to read off the corresponding energy. It is usually the case that the ground state can only be prepared approximately with some overlap γ to the true ground state. Since QPE only has a chance to project onto the true ground state proportional to γ^2, it is often worth trying to improve this overlap. In fact, recent resource estimates show that even though accurate state preparation can be expensive, it is almost always worth doing in realistic scenarios [16].

Specifically, the quantum chemistry problem under consideration is the non-relativistic Born-Oppenheimer electronic Hamiltonian given by:

$$H = -\sum_{i=1}^{N_e} \frac{\nabla_i^2}{2} - \sum_{i=1}^{N_e}\sum_{a=1}^{N_A} \frac{\xi_l}{\|R_a - r_i\|} + \sum_{i\neq j=1}^{N_e} \frac{1}{2\|r_i - r_j\|}, \tag{3.1}$$

where ij index N_e electrons with positions r, a indexes N_A nuclei with atomic charge ξ and position R, and $\|\cdot\|$ denotes the 2-norm. The three terms in the sum are the kinetic energy of the electrons, the nuclear-electron attraction, and the electron-electron repulsion, often denoted T, U, and V, respectively. Working with the Hamiltonian in this form is known as *first quantization*, and the natural parameters to express the wavefunction in are the positions and momenta of the electrons.

However, one can also introduce a basis set (often Gaussians for molecules and planewaves

for materials) and express the Hamiltonian in *second quantization* instead:

$$H = \sum_{ij}^{N_{orb}} h_{ij} a_i^\dagger a_j + \sum_{ijkl}^{N_{orb}} g_{ijkl} a_i^\dagger a_j^\dagger a_k a_l. \tag{3.2}$$

In this formulation, $ijkl$ index N_{orb} molecular orbitals constructed from the basis set, $a^\dagger$ (a) are creation (annihilation) operators, and the h_{ij} and g_{ijkl} coefficients are precomputed as integrals of the T, U, and V terms from Eqn. (3.1) over the molecular orbitals. The convenience of second quantization is that now the natural parameters of the wavefunction are simply the coefficients of the molecular orbitals.

As we will showcase below, the state preparation circuits for first-quantized and second-quantized algorithms can differ in structure significantly. We would like to use application-inspired benchmarks to compare the performance of state preparation circuits in these two competing approaches.

3.1. Antisymmetrization for First-Quantized State Preparation. Recent work [21] has shown that fault-tolerant first-quantized algorithms can scale better than second-quantized algorithms. This is primarily due to the fact that there are often many more molecular orbitals needed in second-quantized approaches than there are actual electrons. However, one downside to moving to first quantization is that the trial state now must be explicitly antisymmetrized. As this is a major difference between first-quantized and second-quantized state preparation, we will focus on building a benchmark for the antisymmetrization circuit.

Antisymmetrization can be done efficiently on a quantum computer [2] with four steps:

1. Prepare an auxiliary register in an even superposition of all possible strings of the target length.
2. Sort this auxiliary register and store the outcome of each comparison.
3. Delete any collisions from the auxiliary register.
4. Apply the reverse of the sort to the target register.

The core routine of the antisymmetrization circuit is the sort, which must be done using a sorting network. An example sorting network is shown in Fig. 3.1 and consists of many (potentially parallel) comparator operations performing inequality tests and controlled-SWAPs.

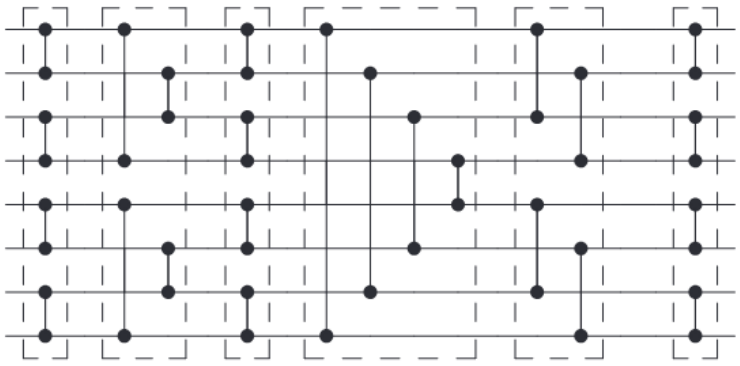


FIG. 3.1. *Example bitonic sort on 8 inputs, which is a major component of the antisymmetrization circuit. Reproduced from [2].*

3.2. Block-Encoded Hamiltonians for Second-Quantized State Preparation. Unlike the first-quantized approach above, the antisymmetrization of the wavefunction in

second quantization is handled by ensuring the proper anticommutation relations of the ladder operators in Eqn. (3.2). Instead, we only have to worry about preparing an accurate trial wavefunction. While there are many strategies for second-quantized state preparation, we will focus here on the near-optimal filter-based approach from Lin and Tong [10]. The key component in the filter-based state preparation approach is "block encoding" of the Hamiltonian in Eqn. (3.2), which we will implement using the linear combination of unitaries (LCU) approach [4]. The LCU approach generally consists of performing controlled rotations to set the h_{ij} and g_{ijkl} coefficients in a "selection" register, performing multicontrolled operations controlled by the selection register, and then unsetting the selection register. An example circuit for the H_2 molecule can be found in Fig. 3.2.

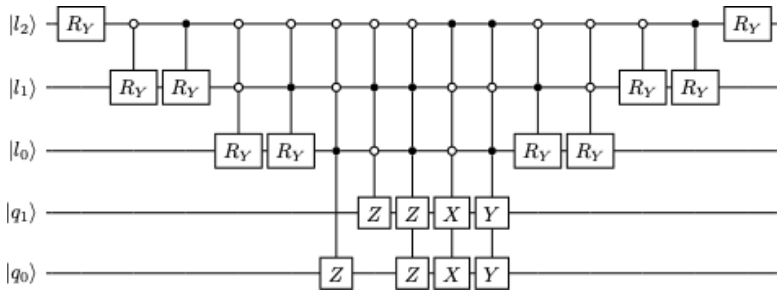
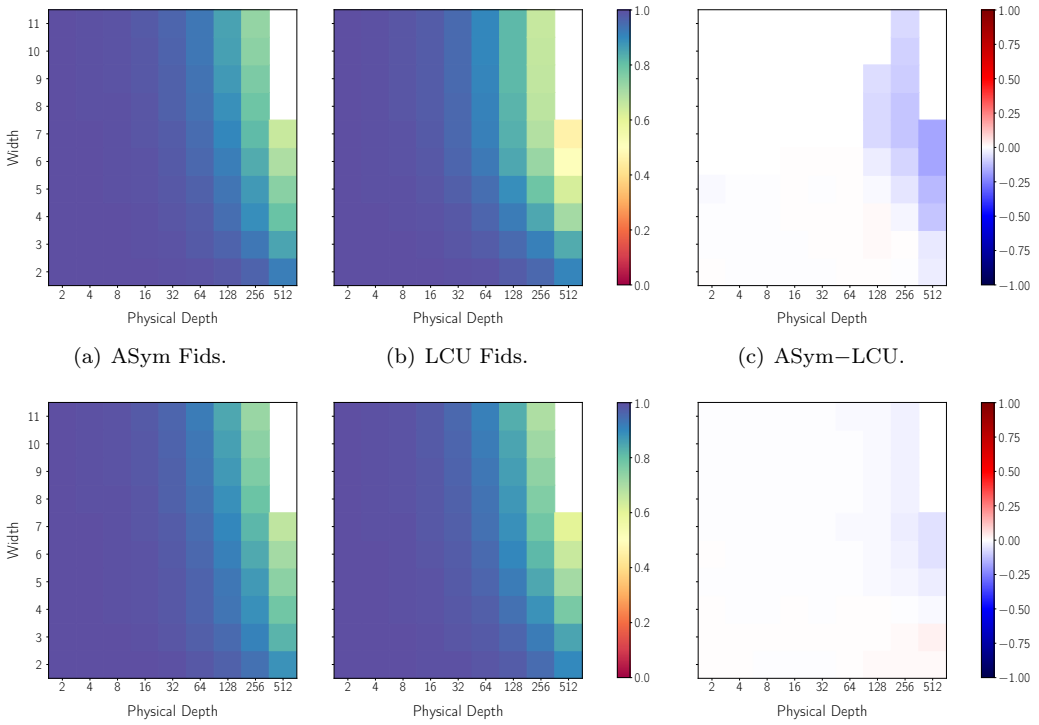


FIG. 3.2. *Exemplar LCU circuit for the tapered H_2 molecular Hamiltonian.*

Unlike the bitonic sort in Fig. 3.1, one can see that the block-encoded Hamiltonian contains significantly more multicontrolled operations and uses the qubit registers much more inhomonogeneously, i.e. there is a clear distinction between the selection register which act as controls and the system register which act as targets.

4. Results and Discussion. In this section, we present numerically simulated results of our benchmarks created for 11-qubit antisymmetrization (ASym) and LCU circuits. Both benchmarks were created using $N_{sc} = 300$ subcircuits per width-depth pair for each width from 2 to 11 and exponentially increasing depths from 2 to 512, excluding widths 8 through 11 for the 512 depth. We used 100 mirror samples per subcircuit and 100 reference circuits for each width. The mirror circuits were simulated using a noise model with one- and two-qubit Hamiltonian and stochastic error generators. The coefficients of the error generators were sampled uniformly from 0 to a maximum strength varied by the error type. The strengths used for the one- and two-qubit Hamiltonian generators were 0.005 and 0.01, respectively, while those for the stochastic error generators were 0.002 and 0.004 for one- and two-qubits.

Figures 4.1(a) and (b) show the mean subcircuit fidelities by subcircuit shape for the ASym and LCU benchmarks. The fidelity serves as an indicator of the circuit's performance, where a circuit executed perfectly will have a fidelity of 1.0, decreasing towards 0.0 as the circuit is executed less faithfully. As the width and depth of the subcircuits approach the width and depth of the full circuit, we expect that the process fidelity of the subcircuits approaches that of the full circuit. Looking to the volumetric benchmarking (VB) plots, we see that the subcircuit fidelities decrease with increasing size in both benchmarks, as we would expect since larger circuits admit more opportunities for errors to occur. Perhaps the more interesting quality of the plots is whether they reveal differing performance between the two algorithms. Figure 4.1(c) shows the difference between the mean subcircuit fidelities of the two benchmarks, from which we observe that the ASym mirror circuits were executed more faithfully than the LCU ones, with the difference becoming increasingly apparent at larger circuit sizes. The greater prominence of the performance differences are likely due



(a) ASym Fids. (b) LCU Fids. (c) ASym−LCU.

(d) Adj. ASym Fids. (e) Adj. LCU Fids. (f) Adj. ASym−LCU.

FIG. 4.1. *Mean mirror circuit process fidelities by width and physical depth for ASym and LCU benchmarks and difference between mean fidelities between ASym and LCU. (a)-(c) use raw fidelities, whereas (d)-(f) use density-adjusted fidelities (denoted Adj.).*

to larger subcircuits capturing more of the structure of the parent circuit, whereas smaller subcircuits are relatively indistinguishable from each other regardless of the circuit they were chosen from. In particular, a 2×2 subcircuit may look similar whether sampled from the ASym or LCU circuit, but a 7×512 subcircuit from one is unlikely to resemble a subcircuit of the same shape from the other.

4.1. Density-adjusted fidelities. Although circuit structure is a plausible explanation for the differences in the subcircuit fidelities between the two benchmarks, there are other relevant factors to consider, namely the gate density of the subcircuits (i.e., the ratio of the gate count to the circuit size). Generally, we would expect that, given two circuits with the same shape and perfect idles but different gate densities, the denser circuit will have a lower fidelity. In real systems, the two-qubit gate density is typically the driver of infidelity since idle and one-qubit gate errors are often significant. As the idle gates are perfect in our numerical simulations, we expect the density to impact the circuit fidelity.

Without accounting for the gate densities, then, we cannot attribute the differences in fidelities between the types of subcircuits to solely structural properties. Moreover, the subcircuits are not guaranteed to have equal densities to their parent circuits, which impacts how representative they are of the full circuit. Attempting to recover a density-agnostic measure of circuit performance, we computed a *density-adjusted fidelity* F_{adj} for each subcircuit to estimate the fidelity of the subcircuit if it had a target density ξ rather than its actual density ξ_{sc}. As a function of the subcircuit's "raw" or unaltered fidelity F_{raw}, the

density-adjusted fidelity is given by

$$F_{adj} = (F_{raw})^{\xi/\xi_{sc}}\,. \tag{4.1}$$

Using these density-adjusted fidelities, we replicated the same VB plots as before, setting the target density to $\xi = 1/8$ as a proxy for the average gate density of the two circuits, as shown in Figs. 4.1(d)-(f). These density-adjusted plots exhibit similar performance differences to those seen in the raw subcircuit fidelities, although slightly damped, indicating the original performance differences were likely artificially amplified by, but not entirely due to, unequal densities between subcircuits of the same shape.

4.2. Predicting full circuit fidelity from subcircuits. For our purposes, whether raw or density-adjusted fidelities are a better metric of performance primarily depends on which ones better align to full circuit performance. To test this, we used MCFE to measure the process fidelity of the full ASym and LCU circuits. Then, we used the benchmarking data to predict the fidelity of the respective full circuit using a simple error rate model, where we assume that the fidelity F_C of the full circuit C with size $s_C = w_C d_C$ can be computed from an "effective" error rate ε_C by

$$F_C = (1 - \varepsilon_C)^{s_C}. \tag{4.2}$$

To predict F_C from a set of w by d subcircuits, we first compute an estimator $\hat{\varepsilon}_{wd}$ of ε_C defined as

$$\hat{\varepsilon}_{wd} = 1 - \mathrm{GM}\left\{F_{wd,1}, \ldots, F_{wd,n}\right\}^{1/s}, \tag{4.3}$$

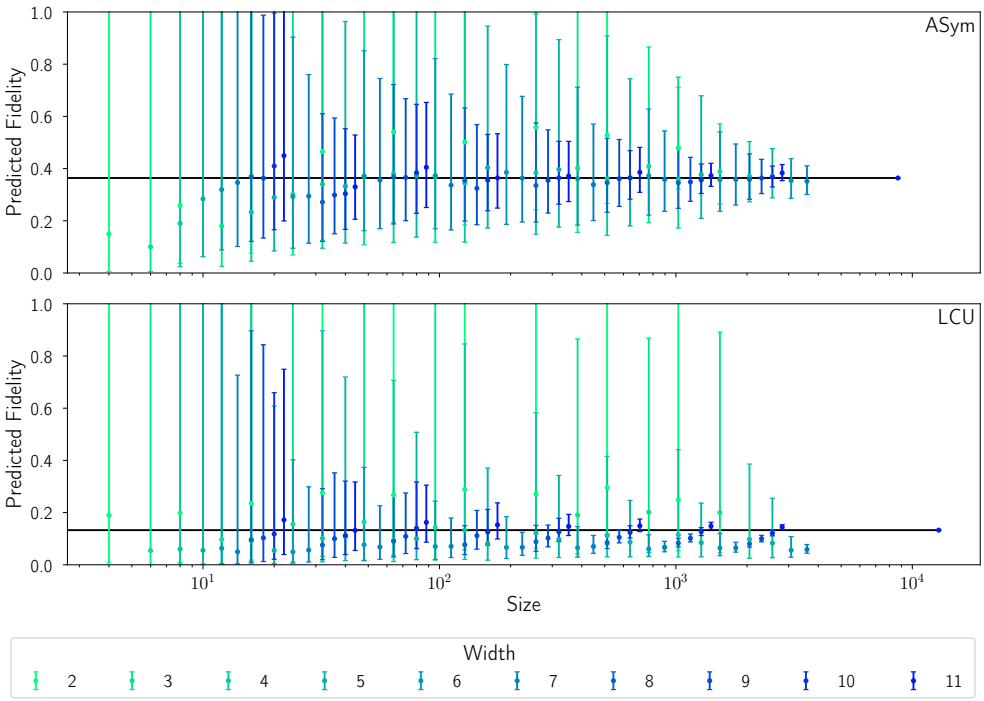
where $s = wd$ is the size of the subcircuits, $F_{wd,i}$ is the fidelity of the ith subcircuit out of the N_{sc} subcircuits of the particular shape, and GM denotes the geometric mean. The predicted fidelity $\hat{F}_{wd}$ is then calculated analogously as

$$\hat{F}_{wd} = (1 - \hat{\varepsilon}_{wd})^{s_C}. \tag{4.4}$$

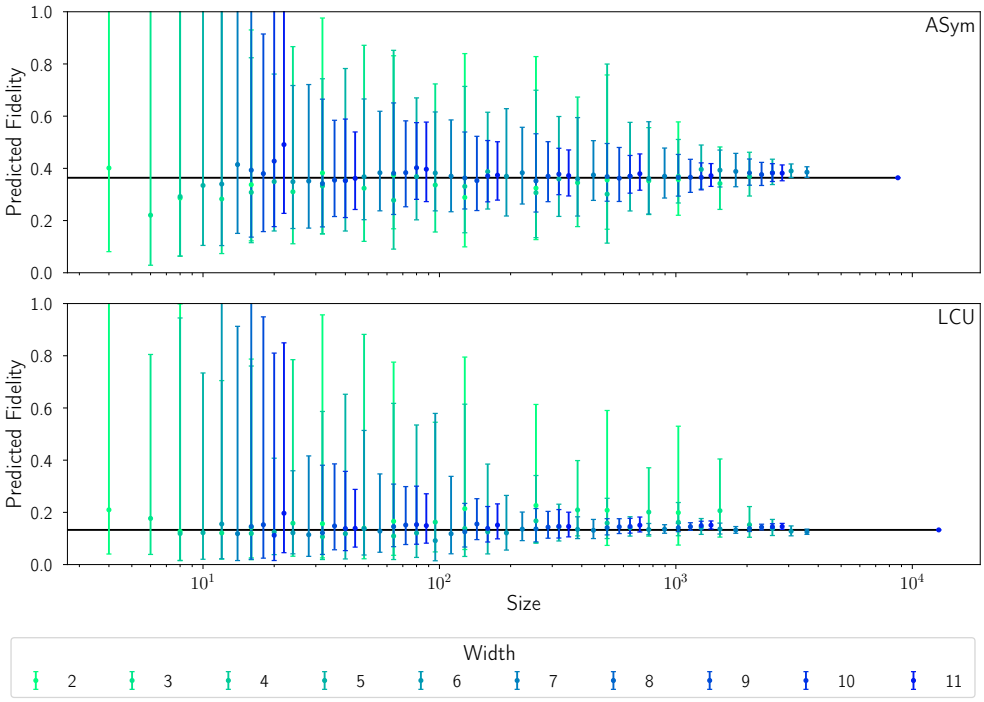
We then tested this model with both the raw and density-adjusted fidelities as shown in Figs. 4.2(a) and (b), respectively. In both figures, the error bars indicate the expected range of fidelities calculated from the geometric standard deviation, which is given by

$$\sigma_{GM} = \exp\sqrt{\frac{1}{N_{sc}}\sum_{i=1}^{N_{sc}}\left(\ln\frac{\hat{F}_{wd,i}}{\hat{F}_{wd}}\right)}, \tag{4.5}$$

where $\hat{F}_{wd,i}$ is the fidelity prediction based on the ith w by d subcircuit. The upper and lower error bars are then given by $\hat{F}_{wd} \cdot \sigma_{GM}$ and $\hat{F}_{wd}/\sigma_{GM}$, respectively (analogous to the arithmetic mean plus or minus the typical standard deviation). Since we wanted to test how well the subcircuits could predict the full circuit fidelity independently for each application, the target density used for computing the density-adjusted fidelities was set to the respective densities of the full circuits rather than a common one. For the ASym circuit, both the raw and density-adjusted fidelity predictions similarly tended toward the measured full circuit fidelity, though the spread of predictions was generally smaller using the density-adjusted fidelities as indicated by the error bars. For the LCU circuits, however, the raw subcircuit fidelities resulted in significantly worse predictions than the density-adjusted fidelities. The predictions using the raw fidelities did not result in an obvious consensus even at large subcircuit sizes, while the predictions using the density-adjusted fidelities show a



(a) Raw fidelities.



(b) Density-adjusted fidelities.

FIG. 4.2. *Full circuit fidelity predictions by subcircuit size (width × depth) for ASym (top) and LCU (bottom) circuits. Predictions using raw fidelities are shown in (a) and those using density-adjusted fidelities are shown in (b). Colors indicate width of subcircuits. Solid black line indicates actual full circuit fidelity.*

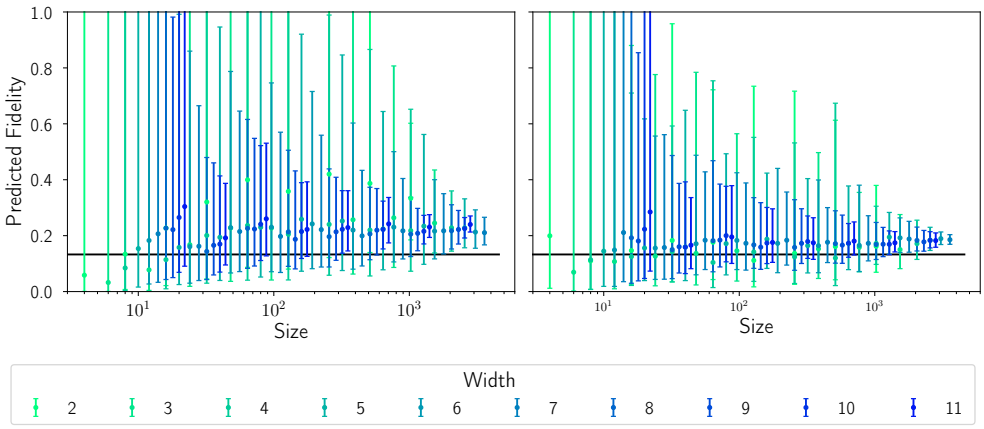


FIG. 4.3. *Full circuit fidelity predictions for LCU ciruit by subcircuit size (width × depth) using ASym subcircuit data. Predictions using raw fidelities are shown on the left and density-adjusted fidelities on the right (target density set to LCU full circuit density). Colors indicate width of subcircuits. Solid black line indicates actual LCU full circuit fidelity.*

much clearer convergence towards the full circuit fidelity. Notably, the mean density of the ASym subcircuits was within 0.5% of the full circuit density, while the mean density of the LCU subcircuits was more than 11% greater than the full circuit density, explaining why the raw ASym predictions worked relatively well whereas the raw LCU predictions were largely inaccurate.

Paying particular attention to the effect of the subcircuit width on the accuracy of predictions, we observe that high-width subcircuits resulted in significantly better predictions than those from low-width subcircuits of similar size, both in terms of the mean predicted value and the spread of predictions. Such results coincide with the well-established phenomenon that measured one- and two-qubit error rates — often used as metrics in periodic device calibration — are poor predictors of real device performance.

4.3. Cross-predicting circuit performance. The relative efficacy of the subcircuits in predicting the full circuit fidelities demonstrates that subcircuit fidelities, particular density-adjusted ones, are a sufficient proxy for measuring the performance of the full circuit. However, one might consider whether they reveal authentic distinctions in a device's capability to run the full circuit due to structural differences in the circuits, or if the different results are simply a consequence of unequal circuit depth and density. If the latter claim were true, we would expect that, using our fidelity prediction model, we could predict the full circuit fidelity of the LCU circuit from the ASym subcircuits, or vice versa. In hopes of rejecting such a possibility, we attempted to do exactly this. Using ASym subcircuit fidelities to find the effective error rate, we computed the same predictions as before, but using the width, depth, and density (where applicable) of the LCU circuit, the results of which are depicted in Fig. 4.3. Using both the raw and density-adjusted fidelities, we see that the predictions do not converge to the full circuit fidelity, implying that we have identified performance differences that cannot be attributed to superficial properties of the circuits.

From this attempt at cross-predicting circuit performance, Fig. 4.3 demonstrates that the ASym data predict a greater fidelity than the LCU circuit actually has; that is, the full circuit fidelity predictions tend towards 0.2 from the ASym data while the true LCU circuit fidelity is closer to 0.15. More quantitatively, we can examine the differences in performance by comparing the effective error rates between the two circuits, given in Table 4.1. For each

TABLE 4.1
Actual (ε_C) and estimated ($\hat{\varepsilon}_{raw}$ and $\hat{\varepsilon}_{adj}$) effective error rates for ASym and LCU circuits.

	ε_C	$\hat{\varepsilon}_{raw}$	$\hat{\varepsilon}_{adj}$
ASym	1.163×10^{-4}	$1.213(276) \times 10^{-4}$	$1.185(126) \times 10^{-4}$
LCU	1.561×10^{-4}	$1.731(350) \times 10^{-4}$	$1.498(133) \times 10^{-4}$
Difference	3.98×10^{-5}	5.66×10^{-5}	3.61×10^{-5}

circuit, the effective error rate was computed as

$$\varepsilon_C = 1 - (F_C)^{1/s_C}, \tag{4.6}$$

where s_C is the size of the full circuit and F_C is the fidelity estimate found using MCFE, which we note is consistent with Eqn. (4.2). To estimate the error rate from the subcircuit fidelities, we computed the error rate for each shape as in Eqn. (4.3) and then took a simple average over those error rates, resulting in the estimated error rates $\hat{\varepsilon}_{raw}$ and $\hat{\varepsilon}_{adj}$, corresponding to estimates using raw and density-adjusted fidelities. It is worth mentioning that a more intelligent scheme could be used to average these error rates as we know that higher width or higher size subcircuits generally yield for accurate predictions; however, a simple arithmetic mean is used here to enable straighforward analysis and error bars.

The results in Table 4.1 coincide with much of what we have already seen in the fidelity predictions, namely that estimates using subcircuit data produce close approximations for the directly-measured full circuit error rates, with higher accuracy using the density-adjusted fidelities. A novel result yields from comparing the error rates for the different circuits, revealing significantly lower error rates for the ASym circuit. Notably, we find that the estimated error rates from density-adjusted subcircuit fidelities closely mirror the actual error rates, even in the relative difference between the two circuits, indicating that subcircuit data can accurately quantify performance differences between application circuits.

5. Conclusions. In this work, we described a general method for creating scalable benchmarks from any application circuit using mirrored subcircuits and applied it to two quantum chemistry application circuits. We showed using simulated results that subcircuit fidelities can accurately predict full circuit fidelities, giving credence to the notion of using an effective error rate as a mechanism for comparing the performance of different circuits. Crucially, we found that the fidelity predictions using subcircuit data are specific to the parent circuit, even when accounting for properties such as the width, depth, and gate density, indicating that our benchmarking scheme is able to distinguish differences in performance resulting from structural properties of the circuits rather than superficial ones.

These results demonstrate that subcircuit techniques for benchmarking provide a scalable proxy for full circuit performance, but several open questions remain. Whether these results can be replicated using a real device instead of a simulated noise model is still unclear. Moreover, our methods for subcircuit selection are imperfect, with the simple method suffering from computational complexity and the connected components method failing to guarantee a sufficient number of subcircuits. We also have yet to examine the differences in results between subcircuit selection methods, which could have significant consequences on how well a generated benchmark mirrors the performance of the application circuit and how we interpret the benchmarking data. Even without regard to subcircuit selection methods, more work is needed to assess the resilience of our fidelity prediction and error rate model, which currently lacks a strong analytical foundation. More sophisticated models may provide more accurate results or allow us to bound the error on our predictions.

REFERENCES

[1] M. Benedetti, D. Garcia-Pintos, O. Perdomo, V. Leyton-Ortega, Y. Nam, and A. Perdomo-Ortiz, *A generative modeling approach for benchmarking and training shallow quantum circuits*, npj Quantum Information, 5 (2019), p. 45.

[2] D. W. Berry, M. Kieferová, A. Scherer, Y. R. Sanders, G. H. Low, N. Wiebe, C. Gidney, and R. Babbush, *Improved techniques for preparing eigenstates of fermionic Hamiltonians*, npj Quantum Information, 4 (2018), p. 22.

[3] R. Blume-Kohout, J. K. Gamble, E. Nielsen, K. Rudinger, J. Mizrahi, K. Fortier, and P. Maunz, *Demonstration of qubit operations below a rigorous fault tolerance threshold with gate set tomography*, Nature Communications, 8 (2017), p. 14485.

[4] A. M. Childs and N. Wiebe, *Hamiltonian Simulation Using Linear Combinations of Unitary Operations*, Quantum Inf. Comput., 12 (2012), pp. 0901–0924.

[5] A. W. Cross, L. S. Bishop, S. Sheldon, P. D. Nation, and J. M. Gambetta, *Validating quantum computers using randomized model circuits*, Physical Review A, 100 (2019).

[6] P.-L. Dallaire-Demers, M. Stęchły, J. F. Gonthier, N. T. Bashige, J. Romero, and Y. Cao, *An application benchmark for fermionic quantum simulations*, 2020.

[7] Y. Dong and L. Lin, *Random circuit block-encoded matrix and a proposal of quantum LINPACK benchmark*, Physical Review A, 103 (2021).

[8] J. R. Finžgar, P. Ross, L. Hölscher, J. Klepsch, and A. Luckow, *QUARK: A framework for quantum computing application benchmarking*, in 2022 IEEE International Conference on Quantum Computing and Engineering (QCE), 2022, pp. 226–237.

[9] A. Li, S. Stein, S. Krishnamoorthy, and J. Ang, *QASMBench: A low-level qasm benchmark suite for nisq evaluation and simulation*, 2022.

[10] L. Lin and Y. Tong, *Near-optimal ground state preparation*, Quantum, 4 (2020), p. 372.

[11] T. Lubinski, S. Johri, P. Varosy, J. Coleman, L. Zhao, J. Necaise, C. H. Baldwin, K. Mayer, and T. Proctor, *Application-oriented performance benchmarks for quantum computing*, 2023.

[12] S. Martiel, T. Ayral, and C. Allouche, *Benchmarking quantum coprocessors in an application-centric, hardware-agnostic, and scalable way*, IEEE Transactions on Quantum Engineering, 2 (2021), pp. 1–11.

[13] A. J. McCaskey, Z. P. Parks, J. Jakowski, S. V. Moore, T. D. Morris, T. S. Humble, and R. C. Pooser, *Quantum chemistry as a benchmark for near-term quantum computers*, npj Quantum Information, 5 (2019), p. 99.

[14] K. Mesman, Z. Al-Ars, and M. Möller, *QPack: Quantum approximate optimization algorithms as universal benchmark for quantum computers*, 2022.

[15] D. Mills, S. Sivarajah, T. L. Scholten, and R. Duncan, *Application-motivated, holistic benchmarking of a full quantum computing stack*, Quantum, 5 (2021), p. 415.

[16] S. Pathak, A. E. Russo, S. K. Seritan, and A. D. Baczewski, *Quantifying t-gate-count improvements for ground-state-energy estimation with near-optimal state preparation*, Phys. Rev. A, 107 (2023), p. L040601.

[17] T. Proctor, K. Rudinger, K. Young, E. Nielsen, and R. Blume-Kohout, *Measuring the capabilities of quantum computers*, Nat. Phys., 18 (2022), pp. 75–79.

[18] T. Proctor, S. Seritan, E. Nielsen, K. Rudinger, K. Young, R. Blume-Kohout, and M. Sarovar, *Establishing trust in quantum computations*, arXiv, (2022).

[19] T. Proctor, S. Seritan, K. Rudinger, E. Nielsen, R. Blume-Kohout, and K. Young, *Scalable randomized benchmarking of quantum computers using mirror circuits*, Physical Review Letters, 129 (2022).

[20] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, *Elucidating reaction mechanisms on quantum computers*, Proc. Nat. Acad. Sci. U.S.A., 114 (2017), pp. 7555–7560.

[21] Y. Su, D. W. Berry, N. Wiebe, N. Rubin, and R. Babbush, *Fault-Tolerant Quantum Simulations of Chemistry in First Quantization*, PRX Quantum, 2 (2021), p. 040332.