

Application-Aware Benchmarking of NISQ Hardware

Joseph Harris^{1,*} and Peter K. Schuhmacher¹

¹*Department of High Performance Computing, Institute of Software Technology,
German Aerospace Center (DLR), Rathaushallee 12, 53757 Sankt Augustin, Germany*

(Dated: October 2024)

Recent experiments have hinted towards an upcoming era of quantum utility, in which quantum hardware is able to outperform classical simulation methods for a variety of real-world applications. In this work, we show how application-inspired families of Clifford circuits can be used to benchmark the capabilities of current hardware for running certain applications, providing a prediction as to how measured expectation value fidelities scale with circuit depth. Considering the specific example of simulating kicked-Ising circuits, we benchmark a 127-qubit superconducting device and suggest how our circuits can also be used to benchmark recent classical simulation methods.

INTRODUCTION

Quantum computers represent a promising new technology due to their potential to provide exponential speedups over classical computers for certain tasks such as the simulation of quantum systems [1–5] or integer factorisation [6].

Despite the rapid evolution of the field, quantum computing remains at a stage where current devices are too small and too erroneous to offer significant computational speedups for various problems when compared to classical algorithms [7]. Recent experiments, however, have hinted towards an upcoming era of quantum utility, in which noisy intermediate-scale quantum (NISQ) hardware is for the first time able to outperform classical simulation methods for a variety of real-world problems [8–10]. Hence, it is necessary to have tools which can be used to benchmark the capability of current devices to run particular applications which have the potential to provide quantum speedups. This requires a full-stack approach, in which areas such as circuit design and compilation for hardware are considered, as well as the leading capabilities of classical simulation techniques.

In this work, we introduce a general framework for producing hardware benchmarks by decomposing target application circuits in terms of Pauli rotation gates. We show how our benchmarking circuits can be chosen to be Clifford and can be orchestrated such that the measurement of a given Pauli string with respect to this circuit is equal to one. Motivated by the recent work of IBM and others [10–17], we consider as a specific example the quantum simulation of a two-dimensional kicked Ising chain followed by an expectation value measurement of a Pauli string. Using our benchmark, we establish a relationship between the fidelity of the measured expectation value and a quantity akin to circuit depth, and show that this relationship is still obeyed when we perform actual kicked Ising simulations. We also suggest that, for specially-chosen circuit parameters, these circuits ought to be more difficult for classical simulation techniques to simulate when compared to the recently developed meth-

ods applied to the original kicked Ising circuits. Thus, we believe they could have use in benchmarking the capabilities of these techniques.

This work is structured as follows. In Section 2, we outline some theoretical background relevant to this work, namely the kicked Ising model, its recent implementation on IBM superconducting hardware and some classical methods recently introduced to simulate it. In Section 3, we describe our general approach for producing benchmarking circuits and consider the particular case of simulating the kicked Ising model using superconducting hardware by benchmarking the `ibm_brisbane` device. In Section 4, we consider the possibility of using our circuits to benchmark those classical simulation methods based on Clifford perturbation or tensor network approaches. Finally, we conclude and discuss the potential for future research.

Alongside this paper, we provide a GitHub repository containing all code and data produced during this work [18].

BACKGROUND

Kicked Ising model

In this work we consider the particular application of simulating the transverse-field (‘kicked’) Ising model with Hamiltonian

$$H = -J \sum_{\langle i,j \rangle} \sigma_z^i \sigma_z^j + h \sum_i \sigma_x^i, \quad (1)$$

where J is the nearest-neighbour coupling strength and h is the strength of a transverse field applied globally. The first sum runs over all pairs of nearest-neighbour qubits (i, j) and the second runs over all qubits in the underlying graph, which we regard to be two-dimensional. The operator σ_z^i , for example, refers to the Pauli- Z operator acting on qubit i .

This choice of application is motivated largely by the recent work of IBM [10], in which this application was

used to showcase the capabilities of their 127-qubit superconducting `ibm_kyiv` processor when combined with heavily tailored error mitigation. The model also has various applications in describing and modelling quantum many-body systems [19–22].

Due to device access restrictions, in this work we instead apply our benchmarks to the near-identical `ibm_brisbane` device, which has the same qubit layout, the same native gate set and broadly similar (but on average slightly greater) error rates. In equation (1), we hence consider our Hamiltonian acting on a two-dimensional heavy-hexagon topology dictated by the qubit layout of these devices (see Figure 1).

In the circuit model, the time evolution e^{-iHt} of H can be simulated using a Trotter decomposition with layers of single and two-qubit gates,

$$e^{-iHt} = [L_1 L_2]^n + \mathcal{O}\left(\frac{t^2}{n}\right) \quad (2)$$

where

$$L_1 = \prod_{\langle i,j \rangle} \exp\left(\frac{Jt}{n} \sigma_i^z \sigma_j^z\right), \quad (3)$$

$$L_2 = \prod_i \exp\left(-\frac{ht}{n} \sigma_j^x\right). \quad (4)$$

In the aforementioned IBM paper, the authors fixed the value of J such that each two-qubit interaction could be implemented using only one native two-qubit gate on the device. This meant that a single L_1 layer could be implemented using only three layers of simultaneous native two-qubit gates; we call these layers $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$ and depict them in Figure 1. They then simulated this model (for varying h) using all 127 physical qubits and a circuit depth of up to 20 Trotter steps (60 layers of CNOT gates). With the careful use of advanced and tailored error mitigation techniques, they were able to extract expectation values of Pauli observables of various weights with good experimental agreement and accuracy. The authors claimed that these results were potentially more accurate than what could be produced using classical simulation techniques, using their own tensor network simulations combined with techniques such as analysis of the observable lightcone to reduce the number of simulated qubits. They showed that this classical data was less accurate than what they had obtained experimentally.

As a result of these claims, several papers were published in which classical simulation techniques were developed or improved upon in order to produce more accurate results than those from the hardware, including references [11–17]. We provide a short overview of these methods in the following subsection.

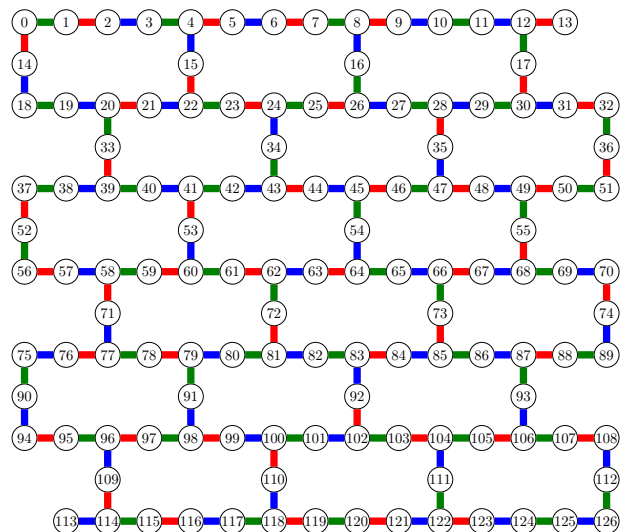


FIG. 1. Qubit layout of the 127-qubit `ibm_kyiv` and `ibm_brisbane` devices, used in the original kicked Ising experiment of [10] and in this work respectively. The colors of the edges indicate the three layers of simultaneous two-qubit gates needed to implement all nearest-neighbour interactions. As in the original work, we apply these layers in the order *red*, *blue*, *green* and call them $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$ respectively.

Classical methods for simulating kicked Ising models

The vast majority of classical simulation methods proposed across the aforementioned papers were tensor network-based simulations [10–12, 14, 16], whereby the expectation value computation is mapped typically to a tensor network contraction problem [23, 24]. These approaches are differentiated by the choice of tensor network ansatz and the method for their lossy contraction, typically relying on identifying ways to simplify computations via analysis of the system’s entanglement structure. They begin to break down when considering highly entangled systems, which have no efficient classical representation, or when capturing non-local correlations.

A further method which we consider uses a technique referred to either as *Clifford perturbation theory* (CPT) or *sparse Pauli dynamics*, originally introduced in [25] and applied to kicked Ising simulations in [12]. This method relies on the fact that Clifford gates can be efficiently classically simulated using the Gottesman–Knill theorem [26], and that a general n -qubit unitary can be decomposed as a series of layers of Pauli rotations,

$$U = U_1(\theta_1) \cdots U_N(\theta_N) \quad (5)$$

where each gate $U_i(\theta_i) = e^{-i\theta_i P_i/2}$ is a Pauli rotation with $P \in \{I, X, Y, Z\}^{\otimes n}$ a Pauli string. The expectation value of a Pauli observable O with respect to this circuit is then

BENCHMARKING NISQ HARDWARE

$$\langle O \rangle = \langle 0^{\otimes n} | U^\dagger O U | 0^{\otimes n} \rangle \quad (6)$$

$$= \langle 0^{\otimes n} | U_N^\dagger(\theta_N) \cdots U_1^\dagger(\theta_1) O U_1(\theta_1) \cdots U_N(\theta_N) | 0^{\otimes n} \rangle. \quad (7)$$

This expectation value can be calculated by iteratively contracting the expression to the middle, using the fact that that

$$e^{i\theta P/2} O e^{-i\theta P/2} \quad (8)$$

$$= \begin{cases} O, & \text{if } [P, O] = 0; \\ \cos(\theta)O + i \sin(\theta)PO, & \text{if } [P, O] \neq 0, \end{cases}$$

where $[P, O] = PO - OP$ for general Pauli strings P and O . This implies that the number of Pauli terms needed to calculate the expectation value grows exponentially with the number of gates which anti-commute with Pauli terms from the evolved observable at each iteration. Since each term is weighted by a product of sines and cosines (and thus each weight is monotonically decreasing with each circuit layer), any small terms are able to be discarded at the expense of increased error.

The final class of simulation methods relevant to our work are those methods which consider the lightcone of measured observables in order to reduce the number of qubits required to perform classical simulations. This idea was used in the original IBM paper [10] and improved upon in [13]. In the latter paper, the authors show that $\langle Z_{62} \rangle$ can be measured accurately on the 127-qubit kicked Ising circuits by considering only up to 31 qubits, within the capabilities of even full state-vector classical simulation. In doing so, they define an *effective fidelity* as the ratio of the experimentally measured expectation value and the ideal value,

$$F_{\text{eff}} = \frac{\text{tr}(\rho O)}{\langle O \rangle_{\text{ideal}}}, \quad (9)$$

where O is the measured observable and ρ is the density matrix of the noisy output quantum state. The authors then define a *effective circuit volume* V_{eff} which governs the scaling of the effective fidelity via

$$F_{\text{eff}} \sim e^{-\varepsilon V_{\text{eff}}} \quad (10)$$

where ε is the dominant error per two-qubit entangling gate. By comparing the mitigated and unmitigated experimental data from [10], the authors showed that some circuits had an effective volume of only around 100 two-qubit gates (compared to 2880 in the original circuit), implying simulation of a smaller circuit could be used to reproduce the data with comparable, if not better, accuracy. They showed this to be the case with good convergence in results as the number of qubits was increased up to a maximum of 31.

One can also compare V_{eff} to V_{lc} , the number of two-qubit gates in the lightcone of O . We will use this quantity later in our work.

Our general approach for producing benchmarking circuits for a specific application circuit and Pauli measurement observable O is the following. We first decompose the gates of the application circuit into single and two-qubit Pauli rotations, and separate the circuit into layers of simultaneously implementable gates. In the kicked Ising case, this is achieved by first writing the circuits in terms of R_X and R_{ZZ} gates (see equations (2)-(4)) – for a single Trotter step, all R_X gates can be implemented simultaneously in a single layer whilst the R_{ZZ} gates require three layers of simultaneous gates to implement all nearest-neighbour interactions (see Figure 1). The ansatz for a benchmarking circuit is then created by duplicating the structure of this circuit while no longer fixing the Pauli associated with each Pauli rotation gate. We also replace each rotation angle with a global angle θ which is the same for all gates in the circuit.

In the case of benchmarking a kicked Ising circuit with L Trotter steps and applied to some connected subset of qubits \mathcal{Q} of the hardware layout shown in Figure 1, this results in an ansatz

$$U(\theta) = \prod_{\ell=1}^L \prod_{j=1}^3 \prod_{\substack{(q_1, q_2) \in \mathcal{D}_j: \\ q_1, q_2 \in \mathcal{Q}}} e^{i\theta P^{(\ell, q_1, q_2)}/2} \prod_{q \in \mathcal{Q}} e^{i\theta P^{(\ell, q)}/2}. \quad (11)$$

For all benchmarking circuits, we fix $\theta = \pi/2$ to yield a Clifford circuit which can thus be efficiently classically simulated using the Gottesman–Knill theorem [26]. This has the natural advantage that at large numbers of qubits we can exactly calculate expectation values. For hardware with a non-parameterised native two-qubit gate acting as the main source of error, we expect the error behaviour of these circuits to be roughly independent of the choice of θ . This is since, for superconducting hardware such as the IBM devices considered in this work, the main source of non-measurement (gate) error is from the two-qubit entangling gates. For both of these devices, the native two-qubit gate is the ECR gate which is fixed and thus independent of our circuit angle parameter θ . In other words, compiling our circuit for this hardware pushes the dependence on θ onto the single-qubit gates, whose error rates are generally several orders of magnitude lower than for the two-qubit ECR gates. This implies that we should expect circuit error behaviour to be broadly insensitive of θ . We would also expect our approach to be useful in the case of a parameterised native two-qubit gate, provided the error rates are roughly independent of the gate parameters.

To fix the individual Paulis in the circuit, we apply an iterative method by moving through the circuit layers in the order in which they are applied. We pick Pauli rotations such that each individual qubit remains in one of the six Pauli ± 1 eigenstates of σ_x, σ_y and σ_z – this allows

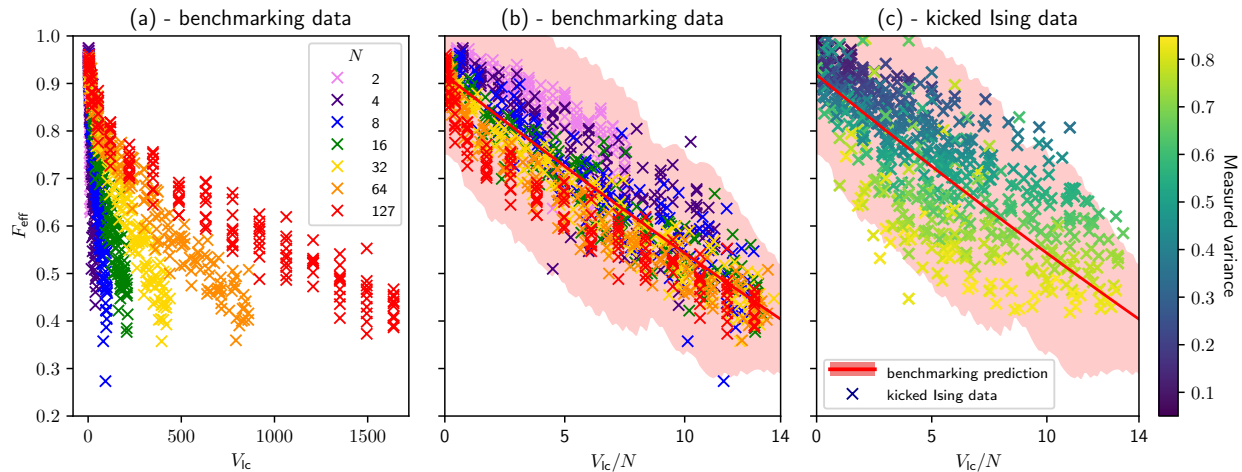


FIG. 2. (a) and (b): the results of running our benchmarking circuits on the `ibm_brisbane` device. We run 10 random benchmarking circuits for each number of qubits $N \in \{2, 4, 8, 16, 32, 64, 127\}$ and each number of layers $L \in \{1, 2, \dots, 15\}$ and in each case plot the effective fidelity F_{eff} of the measured expectation value $\text{tr}(\rho Z_{62})$ (with exact value $\langle Z_{62} \rangle = 1$). In (a), we plot F_{eff} against the exact observable lightcone volume V_{ic} . In (b), we combine this data by instead plotting F_{eff} against V_{ic}/N , enabling us to perform a quadratic curve fitting (red line) with a 3σ error margin (red shading), measured over 100 discrete intervals of V_{ic}/N . (c): the results of running kicked Ising circuits on `ibm_brisbane` and comparison to the benchmarking prediction. In this case we run 900 circuits, where each has a randomly chosen number of qubits $N \in \{1, 2, \dots, 16\}$, circuit layers $L \in \{1, 2, \dots, 15\}$ and single-qubit rotation angle θ_h chosen uniformly at random between 0 and $\pi/4$. For each circuit we calculate $\langle Z_{62} \rangle$ via exact classical simulation, measure $\text{tr}(\rho Z_{62})$ on the device and plot the effective fidelity F_{eff} . We see that almost all the kicked Ising data lies within the 3σ margins of the benchmarking prediction. We also note that the variance in this data is larger, likely since we are measuring expectation values below 1 and thus have reduced measurement precision.

us to keep track of the state $|q\rangle$ of each qubit q in the circuit, since the full circuit state remains a product state. For each two-qubit Pauli rotation $e^{i\theta P^{(q_1, q_2)}/2}$, this can be achieved by picking $P^{(q_1, q_2)} = P^{(q_1)}P^{(q_2)}$ where $|q_1\rangle$ is an eigenstate of $P^{(q_1)}$ and $P^{(q_2)} \in \{\sigma_x, \sigma_y, \sigma_z\}$ is chosen uniformly at random. We then update the state $|q_2\rangle \mapsto P^{(q_2)}|q_2\rangle$, another Pauli eigenstate. For the single-qubit rotations, we simply pick $P^{(q)} \in \{\sigma_x, \sigma_y, \sigma_z\}$ at random and then update the stored state $|q\rangle \mapsto P^{(q)}|q\rangle$. We repeat this procedure until we reach the end of the circuit. At this point, the output state is a known product state $\otimes_q |q\rangle$. We note that we can map any Pauli eigenstate to any other using a single-qubit Pauli rotation gate with a Clifford rotation angle (i.e. a multiple of $\pi/4$). Hence, our final step is to map the output state of the circuit to a $+1$ eigenstate of the Pauli observable O by appending a single-qubit Pauli rotation to each qubit on which O acts non-trivially, with negligible cost to the fidelity. This ensures that with respect to our benchmarking circuits we have $\langle O \rangle = 1$, and thus the effective fidelity F_{eff} is, by equation (9), just the measured expectation value.

In Figure 2, we present the results of a 1050-circuit demonstration in which we benchmark the `ibm_brisbane` device for running unmitigated kicked Ising simulations and measuring the observable $O = Z_{62}$. The measurement of this observable was also considered in the original kicked Ising experiment of [10], and corresponds to

measuring one of the central qubits in the device layout (see Figure 1). Across the demonstration we use increasing numbers of qubits ($N = 2, 4, 8, 16, 32, 64, 127$) and Trotter layers ($L = 1, 2, \dots, 15$), with 10 random circuits per (N, L) combination.

For each random circuit with N qubits and L layers, we first pick uniformly at random a subset of connected qubits $\mathcal{Q}_N \subseteq \{1, 2, \dots, 127\}$ such that $|\mathcal{Q}_N| = N$ and qubit $62 \in \mathcal{Q}_N$. We then start with the ansatz form of equation (11) and populate the gates of the circuit via the random procedure above. We then measure the empirical expectation value $\text{tr}(\rho O)$ using the `ibm_brisbane` device and plot the effective fidelity $F_{\text{eff}} = \text{tr}(\rho O)$. These steps are laid out in the Jupyter notebooks in the associated GitHub repository, alongside the calibration data for the `ibm_brisbane` device at the time of its use [18].

In the middle plot of Figure 2, we see that the fidelity scaling can be made to be almost independent of the number of qubits by considering instead the quantity V_{ic}/N : the size V_{ic} of the observable lightcone (the number of two-qubit gates inside the lightcone of Z_{62}) divided by the number of qubits N in the circuit. This number is comparable to the circuit depth d , but we find it to be more descriptive since d is necessarily quantised to integer values. This allows us to perform a simple curve fitting – we fit the data to a degree-two polynomial and also plot three standard deviations above and below

this curve, measured at regular intervals and indicated by the red shading. This then gives us a prediction for the scaling of the effective fidelity as a function of the circuit depth for circuits with the kicked Ising-type structure.

In the rightmost plot, we test this prediction by running a large number (900) of randomised, non-Clifford kicked Ising circuits. For each circuit, we pick uniformly at random the number of qubits $N \in \{1, 2, \dots, 16\}$, the number of circuit layers $L \in \{1, 2, \dots, 15\}$ and the single-qubit gate rotation angle θ_h uniformly in $(0, \pi/4)$. We classically simulate the exact expectation value $\langle O \rangle$ and then measure the empirical expectation value $\text{tr}(\rho O)$ using the `ibm_brisbane` device. In Figure 2(c), we plot the resulting effective fidelity F_{eff} (see equation (9)), again as a function of V_c/N , and compare these results to the prediction from our benchmarking circuits.

We see that the vast majority of our kicked Ising fidelities lie within the 3σ boundaries predicted by our benchmarking circuits, indicating they are successful in predicting the error behaviour of our application circuits. We note however that the measured variance in the kicked Ising data is higher for greater circuit depths since the measured expectation value tends to decay away from 1 and hence we have reduced measurement precision.

POTENTIAL FOR BENCHMARKING CLASSICAL SIMULATION METHODS

In this section, we show how to generate circuits which have the potential to be useful in benchmarking the capabilities of leading classical methods for simulating kicked Ising models. Specifically, we consider the tensor network and Clifford perturbation techniques outlined in the Background section. We provide some data to support these findings by analysing entanglement entropy.

To do so, we use an algorithm for circuit generation which starts with the same ansatz as in equation (11) and then purposely aims to generate circuits which are hard for Clifford perturbation theory simulation. It does this by picking Pauli rotations which anticommute with the Heisenberg-evolved observable at each layer of the circuit, thus causing its classical description to grow exponentially (see equation (8)). We then show that this class of circuits has greater entanglement entropy on its measured qubits for the measurement of arbitrary weight-two Pauli strings, implying these circuits are more entangling and thus potentially more challenging (but not necessarily impossible) for tensor network simulation techniques to simulate when compared to the kicked Ising circuits of [10].

Our circuits are generated according to the observable O which is to be measured, which here we do not fix. We consider the set S of Pauli strings needed to describe the observable $O = \sum_{P \in S} a_P P$, and track the evolution

of this set in the Heisenberg picture as we move through layers of the circuit (see equation (7)). We start from the deepest layer of the circuit and work backwards, fixing each Pauli rotation gate by brute-force choosing the Pauli which anticommutes with as many elements of S as possible, and then updating the set S using equation (8). After some number of layers (when we expect $|S|$ to be large enough that brute-forcing is no longer feasible), we instead pick Paulis uniformly at random. Since any Pauli has a 50% likelihood of anti-commuting with any other Pauli, we expect $|S|$ to grow by a factor of around $(1.5)^{(|V|+|E|) \cdot (L-L_{\text{bf}})}$ in this region (for a circuit on a qubit graph $G = (V, E)$ with $|V|$ the number of single-qubit gates and $|E|$ the number of two-qubit gates per layer), thus making CPT intractable at larger circuit depths.

With this, we set out the algorithm for circuit generation:

Input: A qubit layout given by connected graph $G = (V, E)$; a Pauli string $O = \prod_{v \in V} O_i$ (where $O_i \in \{I, \sigma_x, \sigma_y, \sigma_z\}$) to be measured; a total number of circuit layers L and some number $L_{\text{bf}} \leq L$ of circuit layers to brute-force. We also input our ansatz based on the target application circuit, in this case given by equation (11). We then pick the corresponding Paulis $P^{(\ell, q_1, q_2)}$ and $P^{(\ell, q)}$ via the algorithm below.

1. Define the set $S = \{O\}$ and $T(P) = \{v \in V \mid P = \otimes_{w \in V} P_w, P_v \neq I\}$, where the $P_w \in \{I, \sigma_x, \sigma_y, \sigma_z\}$; i.e. $T(P)$ is the set of qubits on which P acts non-trivially.
2. For each layer $\ell = L, L-1, \dots, L-L_{\text{bf}}+1$:
 - (a) For each $i \in \{3, 2, 1\}$ and each $(q_1, q_2) \in \mathcal{D}_i$:
 - i. Pick the weight-two Pauli P such that $T(P) = \{q_1, q_2\}$ and the set $\{s \in S \mid [P, s] \neq 0\}$ is as large as possible.
 - ii. Update the set $S \mapsto S \cup \{Ps \mid s \in S, [P, s] \neq 0\}$ and fix $P^{(\ell, q_1, q_2)} = P$.
 - (b) For each $q \in V$:
 - i. Pick the weight-one Pauli P such that $T(P) = \{q\}$ and the set $\{s \in S \mid [P, s] \neq 0\}$ is as large as possible.
 - ii. Update the set $S \mapsto S \cup \{Ps \mid s \in S, [P, s] \neq 0\}$ and fix $P^{(\ell, q)} = P$.
3. For each layer $\ell = L-L_{\text{bf}}, L-L_{\text{bf}}-1, \dots, 1$:
 - (a) For each $(q_1, q_2) \in E$, pick uniformly at random a weight-two Pauli P with $T(P) = \{q_1, q_2\}$ and set $P^{\ell, q_1, q_2} = P$.
 - (b) For each $q \in V$, pick uniformly at random a weight-one Pauli P with $T(P) = \{q\}$ and set $P^{\ell, q} = P$.

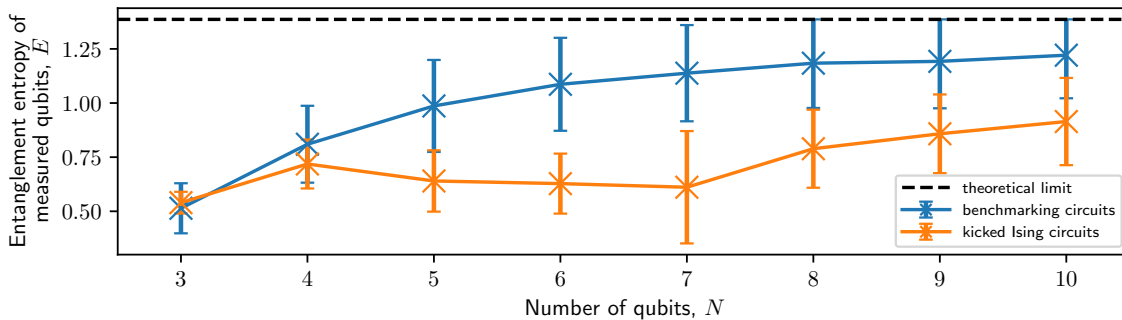


FIG. 3. Comparison between the entanglement entropy growth of our random benchmarking circuits and the kicked Ising circuits of the IBM experiment. Each datapoint is averaged over 1000 random instances with 1σ errorbars. This broadly implies that our benchmarking circuits are more entangling and thus more challenging for tensor network techniques.

We generally find that just one brute-force layer is sufficient for $|S|$ to become intractably large, indicating that this class of circuit should prove more challenging for CPT simulation techniques compared to the original kicked Ising circuits. This is particularly the case when one chooses circuit angle parameter $\theta = \pi/4$ such that terms in expression of the Heisenberg-evolved observable O cannot be excluded, since they all have the same weight (see equation (8)).

We show that this class of random circuit also has the potential to be challenging for tensor network techniques via an analysis of entanglement entropy. In Figure 3, we show the growth of entanglement entropy with the number of qubits in the circuit. Each datapoint is generated using 1000 ‘circuit instances’, where for each instance we pick uniformly at random a connected qubit graph corresponding to a connected subset of the `ibm_brisbane` device. We also pick uniformly at random two measurement qubits q_1, q_2 from this subset and a uniformly random weight-two Pauli observable O to measure on these qubits (i.e. $T(O) = \{q_1, q_2\}$). For our benchmarking circuits, we then generate $U(\pi/4)$ using the algorithm above, classically simulate its evolution and record the entanglement entropy,

$$E = -\text{tr}[\rho_{q_1, q_2} \log(\rho_{q_1, q_2})], \quad (12)$$

where ρ_{q_1, q_2} is the reduced density matrix of the subspace spanned by qubits q_1 and q_2 . For the kicked Ising circuits we repeat the procedure but using standard circuit parameters from [10], namely $\theta_J = -2Jt/n = -\pi/2$ (which enables each two-qubit Pauli- ZZ rotation to be compiled using only a single native two-qubit gate) and $\theta_h = 2ht/n = \pi/4$. We see that the entanglement entropy grows much faster with our benchmarking circuits and appears to plateau closer to the theoretical limit, suggesting that these circuits have application in benchmarking both the CPT and tensor network classical simulation techniques. This result is perhaps unsurprising when one considers that the choice $\theta_J = -\pi/2$ in the

original kicked Ising circuits means that these circuits contain only half as many native two-qubit ECR gates when compiled to the hardware.

CONCLUSION AND FUTURE WORK

In this work, we showed how a novel approach of decomposing application circuits into layers of single and two-qubit Pauli rotation gates can be used to generate benchmarking circuits which are Clifford and accurately reproduce the scaling behaviours of measured expectation value fidelities. We gave a concrete demonstration of this idea by benchmarking the 127-qubit `ibm_brisbane` superconducting device against running kicked Ising demonstrations with various circuit depths, showing good agreement between the observed scaling of the benchmarking and application circuits. Our hope is that this method can be used by researchers to ascertain the capabilities of both existing hardware and new emergent platforms to run particular applications at scale. This is particularly relevant during the NISQ era both for hardware manufacturers seeking to demonstrate the capabilities of their devices as well as end users wishing to optimise an application for a particular hardware platform.

Taking a full-stack approach, we also considered two of the leading techniques for classically simulating kicked Ising models – namely tensor network and Clifford perturbation methods – and gave evidence to show that our circuits may also have application in benchmarking the capabilities of these techniques. The ability to simultaneously benchmark both quantum hardware and classical simulation methods is likewise a useful tool for researchers seeking to demonstrate that we have entered an era of quantum utility. One natural approach for future research could be to produce a comparative benchmark of each of the relevant classical simulation techniques using our benchmarking circuits.

Throughout this research, we have considered only the

effects of running circuits without the use of error mitigation techniques. One natural direction for future research is to consider how these benchmarking circuits could be used to benchmark the capabilities of different error mitigation techniques for different applications. Since we expect the error behaviour of the benchmarking and application circuits to be very similar, the fidelity scaling of a series of error-mitigated benchmarking circuits should form an accurate prediction for the fidelity scaling when running the target application with the same error mitigation. This could also form a valuable tool in comparing the effectiveness of different error mitigation techniques for a single target application.

CODE AND DATA AVAILABILITY

All program code, simulation and measurement results produced over the course of this research are available in the associated GitHub repository [18]. Also available there is the hardware calibration data for the `ibm.brisbane` device available at the time that our circuits were run.

ACKNOWLEDGMENTS

This project was made possible by the DLR Quantum Computing Initiative and the Federal Ministry for Economic Affairs and Climate Action [27].

We thank Michael Epping, Benedikt Fauseweh and Frank Wilhelm-Mauch for fruitful discussions over the course of this work.

We acknowledge the use of IBM Quantum services for this work. The views expressed are those of the authors, and do not reflect the official policy or position of IBM or the IBM Quantum team. All quantum circuits were produced and simulated or submitted to hardware using Qiskit [28].

* joseph.harris@dlr.de

- [1] I. M. Georgescu, S. Ashhab, and F. Nori, Quantum simulation, *Rev. Mod. Phys.* **86**, 153 (2014).
- [2] A. J. Daley, I. Bloch, C. Kokail, S. Flannigan, N. Pearson, M. Troyer, and P. Zoller, Practical quantum advantage in quantum simulation, *Nature* **607**, 667–676 (2022).
- [3] K. L. Brown, W. J. Munro, and V. M. Kendon, Using quantum computers for quantum simulation, *Entropy* **12**, 2268–2307 (2010).
- [4] A. M. Childs, D. Maslov, Y. Nam, N. J. Ross, and Y. Su, Toward the first quantum simulation with quantum speedup, *Proceedings of the National Academy of Sciences* **115**, 9456–9461 (2018).
- [5] C. W. Bauer, Z. Davoudi, A. B. Balantekin, T. Bhattacharya, M. Carena, W. A. de Jong, P. Draper, A. El-Khadra, N. Gemelke, M. Hanada, *et al.*, Quantum simulation for high-energy physics, *PRX Quantum* **4**, 027001 (2023).
- [6] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM Journal on Computing* **26**, 1484–1509 (1997).
- [7] J. Preskill, Quantum Computing in the NISQ era and beyond, *Quantum* **2**, 79 (2018).
- [8] F. Arute *et al.*, Quantum supremacy using a programmable superconducting processor, *Nature* **574**, 505–510 (2019).
- [9] A. J. Daley, I. Bloch, C. Kokail, S. Flannigan, N. Pearson, M. Troyer, and P. Zoller, Practical quantum advantage in quantum simulation, *Nature* **607**, 667–676 (2022).
- [10] Y. Kim, A. Eddins, S. Anand, K. X. Wei, E. van den Berg, S. Rosenblatt, H. Nayfeh, Y. Wu, M. Zaletel, K. Temme, and A. Kandala, Evidence for the utility of quantum computing before fault tolerance, *Nature* **618**, 500 (2023).
- [11] J. Tindall, M. Fishman, E. M. Stoudenmire, and D. Sels, Efficient tensor network simulation of ibm’s eagle kicked ising experiment, *PRX Quantum* **5**, 010308 (2024).
- [12] T. Begušić, J. Gray, and G. K.-L. Chan, Fast and converged classical simulations of evidence for the utility of quantum computing before fault tolerance, *Science Advances* **10**, eadk4321 (2024).
- [13] K. Kechedzhi, S. Isakov, S. Mandrà, B. Villalonga, X. Mi, S. Boixo, and V. Smelyanskiy, Effective quantum volume, fidelity and computational cost of noisy quantum processing experiments, *Future Generation Computer Systems* **153**, 431 (2024).
- [14] S. Anand, K. Temme, A. Kandala, and M. Zaletel, Classical benchmarking of zero noise extrapolation beyond the exactly-verifiable regime (2023), [arXiv:2306.17839](https://arxiv.org/abs/2306.17839).
- [15] Y. Shao, F. Wei, S. Cheng, and Z. Liu, Simulating quantum mean values in noisy variational quantum algorithms: A polynomial-scale approach (2023), [arXiv:2306.05804](https://arxiv.org/abs/2306.05804).
- [16] H.-J. Liao, K. Wang, Z.-S. Zhou, P. Zhang, and T. Xiang, Simulation of ibm’s kicked ising experiment with projected entangled pair operator (2023), [arXiv:2308.03082](https://arxiv.org/abs/2308.03082).
- [17] M. S. Rudolph, E. Fontana, Z. Holmes, and L. Cincio, Classical surrogate simulation of quantum systems with LOWESA, (2023), [arXiv:2308.09109](https://arxiv.org/abs/2308.09109).
- [18] <https://github.com/joeharrisuk/application-aware-benchmarking>.
- [19] X. Mi, M. Ippoliti, C. Quintana, A. Greene, Z. Chen, J. Gross, F. Arute, K. Arya, J. Atalaya, R. Babbush, *et al.*, Time-crystalline eigenstate order on a quantum processor, *Nature* **601**, 531–536 (2021).
- [20] P. Frey and S. Rachel, Realization of a discrete time crystal on 57 qubits of a quantum computer, *Science Advances* **8**, 10.1126/sciadv.abm7652 (2022).
- [21] I.-C. Chen, B. Burdick, Y. Yao, P. P. Orth, and T. Iadecola, Error-mitigated simulation of quantum many-body scars on quantum computers with pulse-level control, *Phys. Rev. Res.* **4**, 043027 (2022).
- [22] X. Mi, M. Sonner, M. Y. Niu, K. W. Lee, B. Foxen, *et al.*, Noise-resilient edge modes on a chain of superconducting qubits, *Science* **378**, 785–790 (2022).
- [23] I. L. Markov and Y. Shi, Simulating quantum computation by contracting tensor networks, *SIAM Journal on Computing* **38**, 963–981 (2008).
- [24] R. Orús, Tensor networks for complex quantum systems,

- [Nature Reviews Physics](#) **1**, 538–550 (2019).
- [25] T. Begušić, K. Hejazi, and G. K.-L. Chan, Simulating quantum circuit expectation values by clifford perturbation theory (2023), [arXiv:2306.04797](#).
- [26] D. Gottesman, The heisenberg representation of quantum computers (1998).
- [27] <https://qci.dlr.de/alqu/>.
- [28] A. Javadi-Abhari, M. Treinish, K. Krsulich, C. J. Wood, J. Lishman, J. Gacon, S. Martiel, P. D. Nation, L. S. Bishop, A. W. Cross, *et al.*, [Quantum computing with Qiskit](#) (2024), [arXiv:2405.08810](#) [quant-ph].