

Improving the efficiency of learning-based error mitigation

Piotr Czarnik^{1,2}, Michael McKerns³, Andrew T. Sornborger³, and Lukasz Cincio^{1,4}

¹Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM, USA.

²Institute of Theoretical Physics, Jagiellonian University, Krakow, Poland.

³Information Sciences, Los Alamos National Laboratory, Los Alamos, NM, USA.

⁴Quantum Science Center, Oak Ridge, TN 37931, USA.

Error mitigation will play an important role in practical applications of near-term noisy quantum computers. Current error mitigation methods typically concentrate on correction quality at the expense of frugality (as measured by the number of additional calls to quantum hardware). To fill the need for highly accurate, yet inexpensive techniques, we introduce an error mitigation scheme that builds on Clifford data regression (CDR). The scheme improves the frugality by carefully choosing the training data and exploiting the symmetries of the problem. We test our approach by correcting long range correlators of the ground state of XY Hamiltonian on IBM Toronto quantum computer. We find that our method is an order of magnitude cheaper while maintaining the same accuracy as the original CDR approach. The efficiency gain enables us to obtain a factor of 10 improvement on the unmitigated results with the total budget as small as $2 \cdot 10^5$ shots. Furthermore, we demonstrate orders of magnitude improvements in frugality for mitigation of energy of the LiH ground state simulated with IBM's Ourense-derived noise model.

1 Introduction

Quantum computers have already demonstrated quantum advantage over classical computers [1, 2]. Nevertheless, current quantum computers suffer from significant levels of hardware noise which limits their computational power [3, 4]. At the same time, high noise levels and limited qubit counts make fault-tolerant quantum computing infeasible [5]. Consequently, to fully utilize the potential of near-term quantum computers, methods to approximately correct the effects of noise without performing quantum error correction are necessary [6, 7]. Such methods are called error mitigation methods [8].

Several approaches to error mitigation have been proposed. One of the most popular error mitigation approaches is Zero Noise Extrapolation, which, using a simple regression model, scales the noise strength in a controlled manner and then extrapolates to the

zero noise limit [6, 9–15]. Another approach is probabilistic error cancellation which introduces additional gates to the executed circuit [9, 16]. The additional gates are introduced probabilistically according to the device noise model in order to cancel noise effects. An alternative approach is to use multiple copies of a noisy state to prepare its purification [17–24]. Such an approach is called Virtual Distillation. Other proposals are based on quantum phase estimation [25], symmetries of the mitigated system [26–29] or are unified methods bringing together different approaches to error mitigation [30–32]. Finally, circuit compilation methods can be used to reduce noise severity in combination with the error mitigation methods mentioned above [33–37]. Although many of these approaches have had success in small circuits, it remains to be seen which of them prove to be the most powerful in the quantum advantage regime.

Another recently proposed approach is learning-based error mitigation [38, 39], whose advantages we demonstrate in this work. Learning-based error mitigation learns the effects of noise from classically simulable quantum circuits which are close to non-simulable circuits of interest. A large class of such approaches evaluates an observable's expectation value for targeted training circuits on a quantum device using training data obtained classically. The training data are fitted with an ansatz capturing a relation between noisy and noiseless expectation values [38, 40–44]. The ansatz is used to correct the noisy expectation value for the circuit of interest. Learning-based error mitigation methods have been shown to perform well for quantum circuits with large qubit counts and depths [38, 45]. They have also been shown to outperform other state-of-the-art approaches [46]. The learning-based approach is a flexible framework which enables one to improve mitigation quality by expanding the training data, for example accounting for the effects of varying noise strength on the expectation values of observables [46, 47]. These features make it a promising candidate for error mitigation for near-term quantum advantage applications.

One of the main limitations of error mitigation is the required shot cost [16]. In the case of learning-based error mitigation it has been shown that to maximize mitigation quality large shot resources are required [46, 47]. The required shot numbers seriously

restrict the practical potential of most powerful and sophisticated approaches. Furthermore, instability in a quantum computer configuration that takes place over time introduces further limitations on the applicability and corrective power of error mitigation. Since quantum computers are effectively experimental apparatuses whose control parameters drift over time, training circuit data acquired over long times using many shots may not accurately reflect noise occurring during the execution of a circuit of interest. We show here that this effect is indeed strong enough to seriously affect error mitigation performance. Finally, some of the most promising near-term applications like variational quantum algorithms require measurement of many quantum circuits to find a final circuit that forms a solution to the problem [8, 48]. To apply error mitigation methods to such problems one needs to optimize the shot-efficiency of the error mitigation approach [49, 50].

In this work, we identify a major problem limiting the shot-efficiency of learning-based error mitigation. One of the most common ways to generate the training data is to use near-Clifford circuits that are similar to the circuit of interest. Such circuits are obtained by substituting non-Clifford gates in the circuit of interest by Clifford gates. We show that such a procedure leads to training circuits for which both noisy and exact expectation values of the observable of interest cluster around 0. We show that this effect leads to a large increase in the shot number necessary to learn the effects of noise on the observable of interest.

We propose two strategies to prevent clustering and improve the shot-efficiency of learning based error mitigation. First, we propose to generate near-Clifford training circuits with widely spread distributions of noisy and exact expectation values of the observable of interest. We propose algorithms to generate such distributions which do not require quantum resources (shots) and therefore are suitable for improving the shot-efficiency.

Second, for circuits with symmetries we propose to use multiple observables, whose expectation values are equal by virtue of the symmetries, to increase the amount of training data available to learn the ansatz for noiseless expectation values. Such an increase enables us to learn from a fraction of the expectation values which are not clustered around 0. We show how to use this strategy to improve the shot-efficiency of error mitigation. We summarize schematically the proposed approaches in Fig. 1. To demonstrate the power of these strategies we mitigate correlation functions of the ground state of the XY model prepared using the IBM Toronto quantum computer. We obtain an order of magnitude improvement in mitigation quality over standard learning-based error mitigation for moderate mitigation shot costs of order 10^5 . Furthermore, for such shot costs, we find that the new method outperforms the standard one even when us-

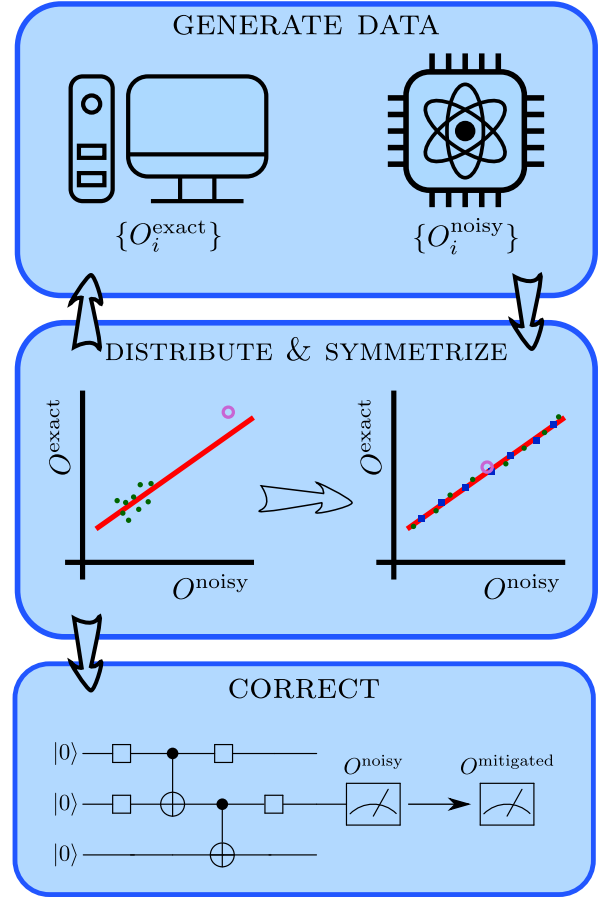


Figure 1: Improved Clifford Data Regression method. Circuit of interest together with observable O are used to generate training data $\{O_i^{\text{exact}}, O_i^{\text{noisy}}\}_i$ made out of observables computed from classically simulable circuits similar to the circuit of interest. They are evaluated on both, classical (O_i^{exact}) and noisy quantum (O_i^{noisy}) computers, as shown in the upper panel. Previous methods for generating training circuits resulted in data clustering, a situation in which the majority of training circuits share similar O^{exact} , as depicted in a cartoon in the left part of the middle panel where the green circles are the training data and $\bullet$ represents $(O^{\text{exact}}, O^{\text{noisy}})$ for the circuit of interest. This effect necessitates the use of large training data sets to maintain high-quality correction $O^{\text{mitigated}}$. We solve that problem by distributing and symmetrizing the training data, as depicted in the right part of the middle panel. That is achieved by generating the training circuits with a large variance of O^{exact} using classical computation. Our approach significantly reduces the shot cost of data-driven error mitigation techniques while preserving their high correction quality. Symmetries of the problem, if present, can be used to enhance the data further, i.e., to generate additional data ($\blacksquare$) from initially generated ($\bullet$) one, at no extra cost. Finally, a learned ansatz (the red line) is used to correct O for the circuit of interest.

ing a factor of 10 fewer shots. As a second example of the new approach's usefulness, we mitigate the ground state energy of LiH molecule that does not have exploitable symmetries. In this case, we obtain orders of magnitude shot cost reduction for a real-device-inspired noise model using the first strategy.

In section 2 we describe a learning-based error mitigation method whose shot-effectiveness we improve here. Section 3 introduces the clustering problem and the strategies to improve shot-efficiency by alleviating it. Section 4 discusses some benchmark results obtained for the IBM Toronto quantum computer and a noise model derived from IBM Ourense by gate tomography. We conclude and discuss possible extensions of this work in Section 5. Technical details of a standard method of near-Clifford training circuit construction on which we improve here are described in Appendix A. In Appendix B we prove the existence of the shot-efficient training circuits for popular variational ansatzes and Pauli observables. There, we also give an efficient semi-analytic procedure to construct a subclass of such circuits. Details of numerical approaches to construct the shot-efficient training data from more general training circuits are given in Appendix C. Appendix D describes the implementation of symmetry constraints on a candidate solution to a minimization problem. Appendix E discusses the IBM Ourense derived noise model. In Appendix F we expand on the choice of hyperparameters for our IBM Toronto error mitigation implementation. In Appendix G we show evidence of the significant impact of hardware changes over time on performance of error mitigation. Finally, in Appendix H, we discuss in more detail the effects of the choice of CDR hyperparameters on the mitigation quality.

2 Clifford Data Regression

Here, we work in the framework of Clifford Data Regression (CDR) [38]. CDR uses classically simulable near-Clifford circuits similar to a circuit of interest to learn a linear ansatz, that is used to mitigate an expectation value of a targeted observable for a circuit of interest. We summarize the method by 4 steps outlined below.

1. Create N_t training circuits with N non-Clifford gates substituting non-Clifford gates in the circuit of interest by nearby Clifford gates. In this work, we additionally allow the non-Clifford gates to be substituted by similar non-Clifford gates to obtain the training circuits. We describe a standard algorithm to generate the training circuits in Appendix A and introduce the shot-efficient training circuit generation methods in Section 3 providing further details in Appendix C.
2. Evaluate an expectation value of the observable of interest O for each of the training circuits both classically and with a quantum computer. With this procedure, one obtains pairs of noisy O_i^{noisy} and exact expectation values O_i^{exact} for each training circuit.

3. Assume a linear relation between O^{exact} and O^{noisy} ,

$$O^{\text{exact}} = aO^{\text{noisy}} + b, \quad (1)$$

where a and b are given by

$$(a, b) = \operatorname{argmin} \sum_{i=1}^{N_t} (O_i^{\text{exact}} - aO_i^{\text{noisy}} - b)^2. \quad (2)$$

4. Use a and b found in step 3 to mitigate the expectation value O , for the circuit of interest

$$O_{\text{coi}}^{\text{mitigated}} = aO_{\text{coi}}^{\text{noisy}} + b. \quad (3)$$

CDR assumes that the correction for the circuit of interest can be learned from classically simulable circuits obtained by substituting some of the gates in the circuit of interest with similar gates. Furthermore, it assumes that the training data are obtained by measuring an expectation value of the observable of interest. To motivate these assumptions, we consider the case of IBM's quantum computers which are one of the leading approaches to digital quantum computing. They have a native gate set built with $R_Z(\theta) = e^{-i\theta/2Z}$, X , $\sqrt{X}$ and CNOT gates, with $\theta \in [0, 2\pi)$, and X , Z being single-qubit Paulis. X , $\sqrt{X}$ and CNOT are Clifford gates, while in general $R_Z(\theta)$ is not a Clifford gate.

Circuits built from Clifford gates and N non-Clifford $R_Z(\theta)$ gates, with N of order of 10 or smaller, can be currently easily simulated classically [51]. Substituting non-Clifford $R_Z(\theta)$ in the circuit of interest with Clifford $R_Z(\theta)$ ($\theta \in \{0, \pi/2, \pi, 3\pi/2\}$), to leave only N non-Clifford $R_Z(\theta)$ gates, we obtain a circuit for which an expectation value of the observable of interest can be computed classically.

At the same time, $R_Z(\theta)$ gates can be performed 'virtually' in a zero-duration mode, consequently being effectively noiseless [52]. More importantly, that implies that noise associated with the gate does not depend on the choice of θ , suggesting that the near-Clifford circuits obtained as sketched above can be used as the training circuits. This conjecture is validated by multiple observations that for IBM's quantum computers, effects of the noise on the observables depend strongly on the locality of the observable of interest and number and arrangement of CNOT gates in the circuit of interest [15, 42, 53], which in the case of CDR are the same for the training data and the circuit of interest.

To learn the correction, an ansatz that connects the noisy and the exact expectation values is needed. For the training circuits that are very similar to the circuit of interest and the training data generated with the observable of interest, we expect that a simple ansatz is sufficient. Indeed, CDR successfully uses a linear ansatz (1). This choice is motivated by an observation that for the IBM's quantum computers, effects of the noise on the observable of interest can

be approximated by an effective global depolarizing noise model [38, 41], which can be perfectly corrected by the linear ansatz (1) [38].

CDR has been demonstrated to provide order-of-magnitude improvements in expectation values of observables for IBM-hardware experiments and diverse applications [38, 45, 54], making it a simple yet powerful approach to IBM-hardware noise mitigation. Below, we discuss the training circuit construction assuming IBM’s native gate set for the sake of presentation clarity.

To apply CDR, one needs to choose a value of the N hyperparameter. As the classical cost of the expectation values simulation grows exponentially with N , $N = 0$ minimizes the classical overhead. Still, simulations of the near-Clifford circuits with $N = 30$ $R_Z(\theta)$ gates can be easily performed with a laptop computer, making minimizing N unnecessary. There are two motivations for using $N > 0$. First, it has been originally proposed [38] to use non-zero N to further bias the training circuits towards the circuit of interest by retaining some of the non-Clifford gates present in the circuit of interest. Along those lines, Ref. [38] reported modest improvements in the mitigation quality with increasing N for a Variational Quantum Eigensolver application and an IBM-quantum-computer-derived noise model.

Another motivation is that $N > 0$ allows a more diverse set of training circuits. Similarly, allowing for non-Clifford R_Z rotations with angles different than in the circuit of interest further increases a pool of potential training circuits. Therefore, we may be able to better characterize variance of deviations from the CDR ansatz (1) among circuits similar to the circuit of interest and leverage that to obtain more robust CDR correction. Here, we use $N = 10$ and $N = 30$ to balance classical computational costs with the benefits of a more diverse set of training circuits. Analogically, though $N_t = 2$ is sufficient to learn a linear ansatz, larger N_t may allow better characterization of the deviations from linear behavior among the near-Clifford circuits. In this work, we explore $N_t \in \{2, \dots, 100\}$, and typically notice systematic improvements of the mitigation quality with increasing N_t as discussed in Section 4.

To place CDR within a context of other learning-based error mitigation methods, we note that several application-specific approaches have been proposed that use classically simulable training circuits specific for a given application [40, 43]. Furthermore, usage of more restrictive training data obtained only from $N = 0$ Clifford circuits has been explored [42]. Additionally, more general sets of noisy training data with expectation values from multiple noise levels have been applied [46, 47], though those methods have been found to be less shot-efficient than CDR [46, 47]. Finally, more general ansatzes like neural networks and random forests have been used to correct multiple ob-

servables and circuits of interests using a single instance of a trained ansatz [43, 44]. We note that such approaches are alternative paths to improve the shot-efficiency of learning-based error mitigation [44].

3 Improving efficiency of learning-based error mitigation

3.1 Clustering of near-Clifford training circuits

The most popular method for choosing training circuits for CDR is to randomly replace most of the non-Clifford gates in the circuit of interest by nearby Clifford gates [38, 55, 56], as discussed in detail in Appendix A. We observe that for circuits of interest with a large number of non-Clifford gates this method results in the clustering of training data caused by the clustering of O^{exact} around 0. We visualize this effect in Fig. 2(a) for a typical training set obtained with a random substitution method for a half-chain correlator of the ground state of an 8-qubit XY model using a noise model derived from gate tomography of IBM Ourense quantum computer.

As a thought experiment we consider the case of both training circuits and circuit of interest satisfying the CDR ansatz (1) perfectly. Then, the clustering of O^{exact} around 0 implies also clustering of O^{noisy} around b . Consequently, a small number of training circuits results in similar values of O^{noisy} . Therefore, when we estimate O^{noisy} from a finite shot measurement, a much higher number of shots is required to distinguish them than in the case when they are not clustered around a single value. If the number of shots is not large enough, coefficients of the fitted CDR ansatz are determined randomly by the shot noise making learning the correct CDR ansatz impossible. We find that this toy model explains the poor quality of CDR error mitigation for the case of the 8-qubit XY model and a typical standard training set obtained with small number of shots (2000 in this example). As we show in Fig. 2(b), error bars of the finite shot estimates of O^{noisy} are so large that one can obtain nearly arbitrary CDR fit coefficients.

3.2 Distributing the training circuits

Despite the observed clustering of O^{exact} , near-Clifford training circuits with large $|O^{\text{exact}}|$ can usually be obtained. As shown in Appendix B, for a circuit of interest compiled to the native IBM’s gate set and an observable of interest being an element of Pauli group, a $N = 1$ training circuit with a single non-Clifford $R_Z(\theta)$ gate and an arbitrary $-1 \leq O^{\text{exact}} \leq 1$ can be found, as long as there is at least one $R_Z(\theta)$ such that O^{exact} depends non-trivially on θ . Additionally, if such a training circuit exists, it can be found with a cost that scales linearly with the number of the Clifford gates. Also, Appendix B shows that such

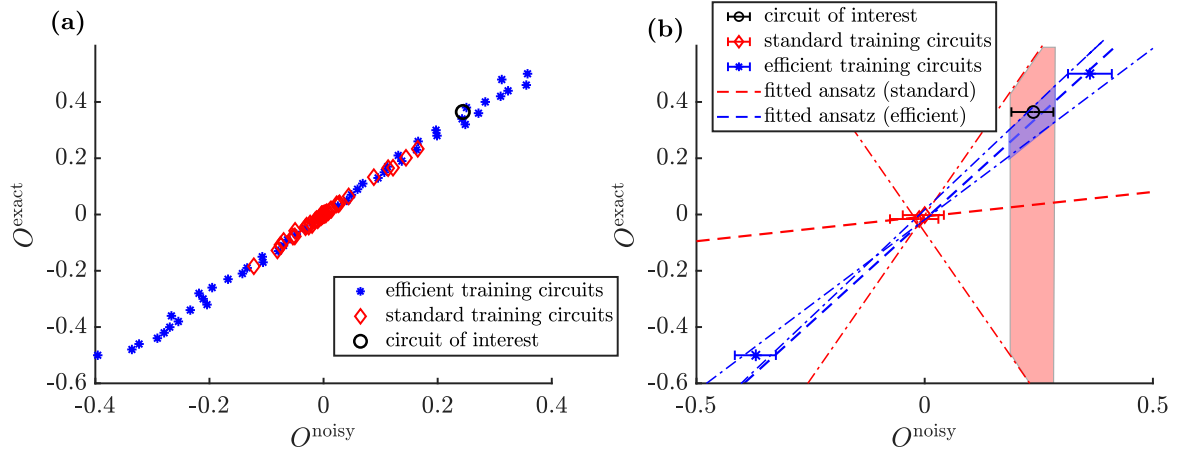


Figure 2: **Improving CDR shot efficiency by distributing training circuits.** In (a) we show O^{exact} versus noisy O^{noisy} expectation values of a given observable computed from a circuit of interest as well as from CDR training circuits. The noisy expectation values were obtained in the limit of infinite shot number. We show results for two methods of training circuit construction. The results obtained with the standard method (random substitutions of non-Clifford gates by Clifford ones) are shown with red diamonds. They cluster around a $O^{\text{noisy}} = O^{\text{exact}} = 0$. The blue asterisks are results obtained for a set of training circuits with a wide distribution of O^{exact} generated by MCMC. This set also prevents clustering of O^{noisy} . In (b) we show the results of CDR error mitigation for a small shot cost. We use two training circuits generated by both methods. We evaluate 1000 shots to estimate O^{noisy} for both training circuits and the circuit of interest. Error bars correspond to 5th and 95th quantiles of O^{noisy} estimated from 100 shot noise instances, while the markers correspond to the median value. We performed CDR fits for all shot noise instances. We show CDR fits corresponding to 50th (dashed lines), 5th and 95th (dashed-dotted lines) quantiles of the mitigated expectation value. The shaded areas show confidence intervals of the mitigated expectation value based on statistical distribution of the fits. We see that CDR with distributed training circuits results in both smaller uncertainty and better quality of the mitigated expectation values. Thus, these results demonstrate that preventing clustering of the training data indeed improves the shot-efficiency of CDR error mitigation. Here, in both (a) and (b), the circuit of interest approximates the ground state of an 8-qubit XY model (7) and $O = X_1 X_5$, where X is a Pauli matrix. To generate O^{noisy} , we use in both panels a noise model described in Appendix E.

circuits can be constructed for popular ansatzes as hardware efficient ansatz and Quantum Approximate Optimization Algorithm (QAOA) ansatz, as long as Pauli observable of interest is not an identity.

Based on this result, to improve the shot-efficiency of CDR, we propose to use training circuits with a widely spread distribution of O^{exact} that has a significant variance. Here, for a Pauli observable of interest, we use a uniform distribution of O^{exact} with variance in $[0.25, 0.5]$. We find that this approach indeed significantly improves the quality of the CDR correction for modest shot requirements as shown in Fig. 2(b) and demonstrated by extensive benchmarks in Section 4.

Several approaches can be used to generate well-distributed training data. First, one can optimize angles of $R_Z(\theta)$ gates under a constraint of N non-Clifford angles to minimize the distance between O_i^{exact} and its target value. We present this approach in detail in Appendix C.1. For a brick wall arrangement of L layers of CNOTs decorated with random single-qubit gates and L proportional to the qubit number Q , in Appendix C.1, we show numerically that this approach requires computational time scaling linearly with the gate number and quadratically with Q . For this setup, in Appendix C.1, we generate the distributed training circuits for $Q = 455$,

$L = 91$ circuits of interest with tens of thousands of CNOTs and single-qubit gates within a few hours of CPU time.

Second, one can use Markov Chain Monte Carlo (MCMC) sampling of near-Clifford circuits to generate a desired O^{exact} distribution. We describe an MCMC algorithm in detail in Appendix C.2. Furthermore, one can use a semi-analytic approach of Appendix B to efficiently generate $N = 1$ training circuits with a cost scaling linearly in the number of Clifford gates. In principle, one can also employ the standard random substitution algorithm supplemented by a postselection. Namely, one could generate near-Clifford circuits randomly substituting non-Clifford $R_Z(\theta)$ in the circuit of interest and accept them as training circuits when they have O^{exact} close enough to the desired one. Because random near-Clifford circuits tend to cluster, a large number of these randomly generated circuits would be needed to find those with the desired distributed observable, which can then be used to generate training data efficiently with fewer shots. Therefore, this approach is inefficient. It is worth noting that none of these approaches requires quantum resources.

Finally, we note that the original CDR proposal [38] considered the case of training circuits with a distribu-

tion of an observable imposed by MCMC sampling of near-Clifford circuits. Nevertheless, both the motivation for that and the distribution were different than the ones considered here. Specifically, Ref. [38] considered a special case of a circuit of interest obtained by minimization of a cost function of a Variational Quantum Algorithm. To further bias the training circuits towards such a circuit of interest and maximize the correction quality in the large shot number limit, the approach in Ref. [38] used near-Clifford circuits that minimize the cost function.

3.3 Enforcing symmetries of the system

Another strategy to improve shot efficiency in the presence of clustering is to exploit symmetries of the mitigated circuit. For certain types of symmetric systems, like those with translational symmetry, multiple observables have the same noiseless expectation values. Furthermore, multiple expectation values of such observables can typically be obtained from a single measurement of the circuit. Consequently, in such a case, one can build training data for multiple symmetric observables using the same number of shots as for a single observable. In the presence of such symmetries, we propose to correct all symmetric observables with CDR while enforcing the constraint that all mitigated expectation values are the same. Using expectation values of many symmetric observables, we increase the probability of including data points with large $|O^{\text{exact}}|$ and $|O^{\text{noisy}}|$ in our training data. The presence of training data with large $|O^{\text{noisy}}|$ improves the shot efficiency of CDR-based error mitigation. By enforcing the constraint, we ensure that such training data affects all mitigated expectation values.

An example of a circuit to which this approach can be applied is a circuit approximating an eigenstate of a translationally invariant Hamiltonian. For example, in the case of a translationally invariant one-dimensional Hamiltonian with periodic boundary conditions, noiseless observable expectation values

$$X_i X_{i'}, \quad i' = (i + l) \bmod Q, \quad i = 1, \dots, Q$$

are the same because of symmetry. Here, Q is the number of qubits, l is an integer and X_i is a Pauli matrix acting on qubit i . At the same time, for IBM quantum hardware, such expectation values can be measured simultaneously. Therefore, simultaneous CDR error mitigation is possible without requiring additional shot cost.

We remark that circuits with symmetries can be obtained through ansatz respecting symmetries of a problem of interest, including widely-used Quantum Alternating Operator Ansatz [57] and Hamiltonian Variational Ansatz [58]. Furthermore, systematic procedures to construct symmetric ansatz for both Variational Quantum Eigensolver and Quantum Machine Learning applications have been proposed [59–61].

More formally, we consider here the case of M observables with equal exact expectation values due to a symmetry of the mitigated system

$$O_1^{\text{exact}} = O_2^{\text{exact}} = \dots = O_M^{\text{exact}}.$$

We propose to create training data by measuring noisy expectation values of all observables for N_t training circuits and classically computing their corresponding exact expectation values

$$\{(O_{ji}^{\text{exact}}, O_{ji}^{\text{noisy}})\}, \quad i = 1, \dots, N_t, \quad j = 1, \dots, M.$$

In a more general case one can also have different sets of training circuits for each observable (N_t may depend on observable index j).

We propose to find mitigated expectation values of these observables by fitting M linear ansatz

$$O_j^{\text{exact}} = a_j O_j^{\text{noisy}} + b_j, \quad j = 1, \dots, M \quad (4)$$

to the training data. Their coefficients a_j, b_j are found by minimizing the cost

$$C = \sum_{ij} (O_{ji}^{\text{exact}} - a_j O_{ji}^{\text{noisy}} - b_j)^2 \quad (5)$$

under the constraint that the mitigated expectation values are equal:

$$a_1 O_{\text{coi},1}^{\text{noisy}} + b_1 = \dots = a_M O_{\text{coi},M}^{\text{noisy}} + b_M. \quad (6)$$

Such a constrained optimization can be performed with the mystic package [62, 63], as in Appendix D. We show an example of the shot-efficiency improvement with this approach in Fig. 3.

While in this section we concentrate on application of symmetric CDR to improve shot-efficiency of error mitigation, it can be also used to enforce symmetries of the mitigated circuit for the mitigated expectation values. Such symmetries are usually broken by the noise, therefore the standard CDR is not guaranteed to restore them.

4 Benchmark results

4.1 The ground state of an XY model

4.1.1 The setup

We test the proposed methods for the ground state of a one-dimensional XY model with periodic boundary conditions given by the Hamiltonian

$$H = \sum_{\langle i,j \rangle} X_i X_j + Y_i Y_j. \quad (7)$$

Here X_i and Y_i are Pauli matrices acting on qubit i . This Hamiltonian is translationally invariant and it preserves the Hamming weight, i.e. it has a $U(1)$ symmetry.

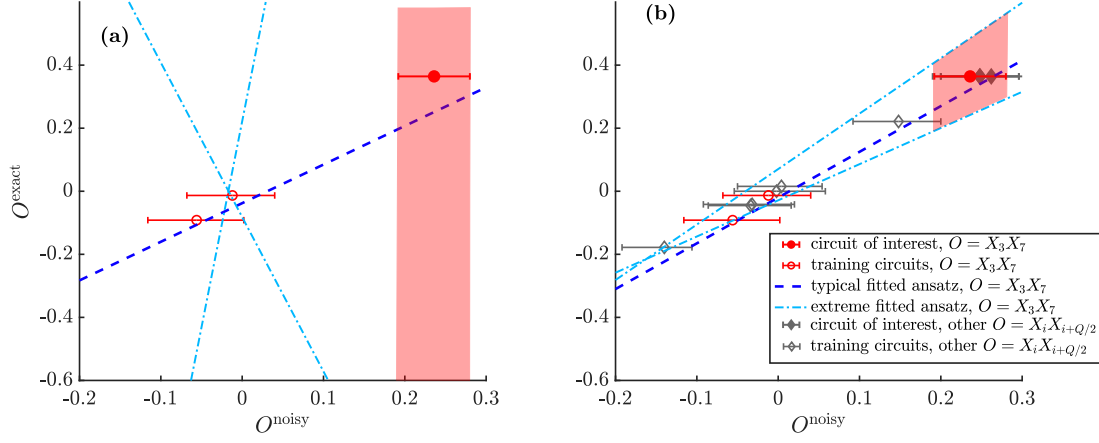


Figure 3: Improving CDR shot efficiency by enforcing symmetries of the mitigated circuit. We perform CDR error mitigation for the ground state of 8-qubit XY model with periodic boundary conditions simulated with the IBM Ourense noise model. We mitigate the expectation value of $O = X_3X_7$ for the ground state using two training circuits generated by the standard substitution algorithm and performing 1000 shots for each circuit. In (a), we show results of standard CDR error mitigation. As in Figure 2(b), we find that expectation values of the training circuits (red circles) cluster around (0,0) and that due to the clustering, the uncertainty of the mitigated expectation value shown by the red shaded area is large. We visualize this effect performing CDR error mitigation for 100 instances of shot noise. (The shot noise is represented by the error bars of O^{noisy} in the same way as in Figure 2(b)). We show typical and extreme CDR fits (defined as in Figure 2(b)) with the blue line and the cyan lines, respectively. In (b), we take advantage of the translational invariance of the ground state which implies that X_1X_5 , X_2X_6 , X_3X_7 , X_4X_8 all have the same noiseless expectation values. We measure all of these observables for both our circuit of interest and the training circuits. They are mitigated simultaneously imposing equality of their mitigated expectation values. The additional observables (induced by symmetry) are shown as gray diamonds in the plot. We correct 100 different shot noise instances of the noisy expectation values and show the fits corresponding to typical and extremal mitigated expectation value of X_3X_7 . We find that uncertainty of the mitigated expectation value and quality of the typical CDR fit are significantly improved with respect to (a). For the sake of transparency we do not show the CDR fits for other observables. They have similar quality to the ones shown here.

We use the Variational Quantum Eigensolver [64] to prepare the ground state for $Q = 6, 8$ lattice sites (or required qubits) with a noiseless classical simulator and enforced translational symmetry. We employ a hardware-efficient ansatz built out of layers of general nearest-neighbor 2-qubit unitaries with a sufficient number of layers (4 for $Q = 6$ and 6 for $Q = 8$) such that the energy of the prepared state matches the exact ground state energy with accuracy better than 10^{-13} . The obtained circuits are compiled using the IBM native gate set.

Here, we reduce the error in the expectation values of long-range correlators for the ground state of the model. Because of the symmetries of the ground state, we have N identical half-chain correlators:

$$\begin{aligned} O_j &= X_j X_{Q/2+j}, & 1 \leq j \leq Q/2, \\ O_j &= Y_{j-Q/2} Y_j, & Q/2 + 1 \leq j \leq Q, \end{aligned} \quad (8)$$

with equal noiseless expectation values. Below we mitigate the errors of expectation values of these observables distributing the training circuits and using the symmetry properties of the model. We note that the $Q = 6$ ($Q = 8$) circuit compiled to IBM's native gate set has 51 (144) CNOTs, with up to 48 (132) CNOTs in a causal cone of a mitigated observable, 20 (32) non-parallelizable layers of CNOTs, and 150 (185) non-Clifford $R_Z(\theta)$ gates. The causal cone of a

mitigated observable contains all the gates affecting its expectation value and indicates error mitigation hardness for local noise [53].

4.1.2 Implementation on quantum hardware

We mitigate half-chain correlators (8) of the $Q = 6$ XY model (7) using IBM's Toronto quantum computer. As a reference, we also perform CDR correction with a standard algorithm described in Section 2 and Appendix A using training circuits obtained via random Clifford substitutions. We use a shot-efficient CDR method combining both approaches described in Section 3. Namely, we perform symmetric CDR for all O_j in (8) following the algorithm described in Section 3.3. Each symmetric observable O_j requires generating independent N_t^j training circuits with widely spread O_j^{exact} . These training circuits are generated using the MCMC algorithm described in Appendix C.2. We describe the construction of the training sets for both approaches in more detail in Appendix F. The shot-frugal approach used here improves CDR shot-efficiency through spreading O_j^{exact} , while symmetric CDR restores uniformity of the mitigated correlators. One may consider generalizations of the applied algorithm which would allow one to generate training circuits with distributions of sev-

eral O_j^{exact} being widely-spread simultaneously. Such training circuits would improve shot-efficiency of the current approach while increasing classical computational cost of the method. We leave their investigation to future work.

We compare the performance of the new and standard approaches using IBM’s Toronto quantum computer. We note that here, we do not use dedicated measurement error mitigation. We use $N_s = 10^4$ shots per circuit measurement and compare the performance of both approaches for a total number of training circuits $N_t \in \{2, 6, 8, 12, 18, 30\}$. For each value of N_t we generate 10 different, independent instances of sets of training circuits. The number of non-Clifford gates is set to $N = 30$, while the circuit of interest contains 150 non-Clifford gates. We compare the performance by computing the absolute error of the mean of the observable expectation values for each set of training circuits

$$\text{absolute error} = \left| \sum_j (O_j^{\text{noisy/mitigated}} - O_j^{\text{exact}}) \right|. \quad (9)$$

For both methods, we compare mean and maximal absolute errors obtained for a given N_t from a sample of 10 sets of training circuits. Furthermore, for each set of training circuits we perform independent measurements of the circuit of interest. We gather the results in Fig. 4 and plot them versus total shot cost of the mitigation, N_s^{tot} .

Figure 4 shows data acquired with standard and efficient CDR methods. We concentrate on the realistic regime of $N_s^{\text{tot}} \in [3 \cdot 10^4, 3.2 \cdot 10^5]$ and show maximum and mean errors obtained with both methods. We observe that the new efficient CDR method displays better performance as well as faster convergence with increasing N_s^{tot} . These results show a substantial improvement of up to an order of magnitude for the new, efficient method. Importantly, efficient CDR reaches its correction capabilities already at small shot budgets. This makes it a preferable tool for applications that require the execution of many different quantum circuits. This is a typical situation in many approaches involving variational quantum algorithms.

4.1.3 Quantum computer noise model

We perform a more detailed benchmark using a noise model obtained by performing full process tomography of basic gates available on IBM’s Ourense quantum computer. Namely, we additionally examine the effect of varying N_s on the performance of shot-efficient CDR. Furthermore, we analyze larger Q and gather better statistics using larger samples of training circuits. The noise model is described in detail in Appendix E. Taking into account that the device is an early generation, already retired quantum computer and that we mitigate circuits with larger Q and depth than in the case of our real-hardware implementation, we reduced gate error rates in the model by a

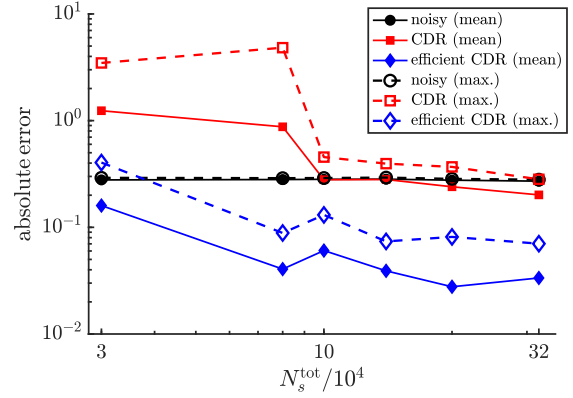


Figure 4: **Benchmark results for IBM Toronto.** We compare standard and efficient CDR performance for the ground state of the $Q = 6$ qubit XY model (7) by mitigating symmetric half-chain correlators (8) of the ground state. The comparison is performed for $N_s = 10^4$ shots per circuit measurement and $N_t \in \{2, 6, 8, 12, 18, 30\}$ training circuits. For each N_t and each CDR version we construct 10 independent sets of training circuits, see details in Section 4.1.2 and Appendix F. Furthermore, for each set we carried out independent measurements of the circuit of interest. For each training set we perform error mitigation and quantify its quality by computing absolute error of the correlators (9). For each method and each N_t value we determine the maximal and the mean absolute error of the mitigated correlators for the sample of training circuits. We plot these errors versus total shot cost N_s^{tot} . For a reference we also plot the errors of unmitigated, raw results (black lines). We observe that shot-efficient CDR displays up to an order-of-magnitude correction improvement over the standard method and faster convergence of the mitigated observables with increasing N_s^{tot} .

factor of 10 with respect to the original Ourense noise model. This allows us to perform the analysis for noise strengths for which basic CDR provides good quality of error mitigation. While current quantum computers may have noise strengths larger than assumed here, we expect that improvements in quantum computing architectures will reduce the error rates in the near-future.

As in the case of our implementation on IBM’s Toronto processor, we mitigate symmetric half-chain correlators (8) of the ground state of the XY model with periodic boundary conditions (7). We choose $Q = 8$. We compare the performance of shot-efficient and standard algorithms using the same set-up as for the IBM Toronto benchmark in Section 4.1.2. This time we consider three different values of $N_s \in \{10^3, 10^4, 10^5\}$ and a larger range of $N_t \in \{2, \dots, 100\}$. Such choice enables us to test the shot-efficient method for large error mitigation total shot costs. Additionally, for each value of N_t and for each method, we generate 50 independent sets of training circuits instead of 10 to decrease finite sample effects. It is currently not possible to perform those extended experiments on real quantum computers due to hardware drift. We discuss those issues in more de-

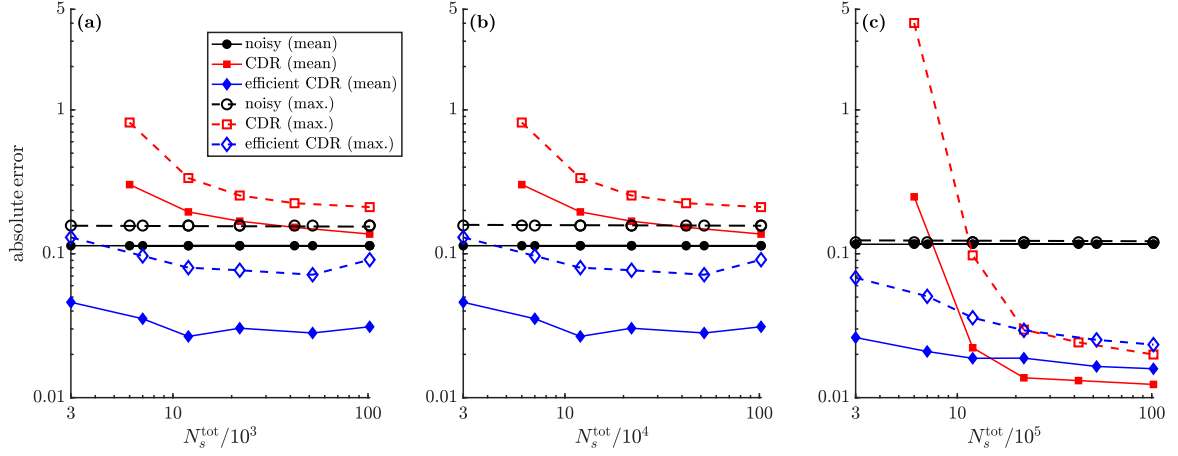


Figure 5: **Benchmark results for a noise model.** Comparison of the standard (red curves) and efficient (blue curves) CDR for the ground state of $Q = 8$ qubit XY model (7) by mitigating symmetric half-chain correlators (8) of the ground state. We use a noise model described in detail in Appendix E. The performance is compared for $N_s = 10^3$ (a), $N_s = 10^4$ (b) and $N_s = 10^5$ shots (c) per a circuit measurement and $N_t \in \{2, \dots, 100\}$ training circuits. For each N_t and each CDR version we construct 50 independent sets of training circuits. They are obtained in the same way as for IBM Toronto error mitigation, see Section 4.1.2. The circuit of interest is measured independently in each set. We plot mean and maximal absolute errors (9) of both CDR methods and the noisy data versus total shot cost of the error mitigation N_s^{tot} . We observe that the shot-efficient version significantly outperforms the standard one for small and moderate total shot costs $N_s^{\text{tot}} = 3 \cdot 10^3$ to $N_s^{\text{tot}} = 1.2 \cdot 10^6$.

tail in Appendix G. Effects caused by hardware drift further motivate shot-efficient error mitigation techniques.

We gather the results (the mean and the maximal errors evaluated from a sample of 50 sets of training circuits) and plot them versus N_s^{tot} in Figure 5. Results obtained with $N_s \in \{10^3, 10^4\}$ resemble those observed on the real device. Namely, even in the limit of large N_t we see that the shot-efficient algorithm outperforms standard CDR algorithm. At the same time, we notice that the shot-efficient mitigated results converge much faster with increasing N_s^{tot} than the standard mitigated ones. For $N_s = 10^5$ we again see much faster convergence of the shot-efficient CDR which outperforms the standard version for $N_s^{\text{tot}} \leq 1.2 \cdot 10^6$. Standard CDR slightly outperforms the shot-efficient version for the largest $N_s^{\text{tot}} \geq 2.2 \cdot 10^6$.

Overall, we observe that for small and moderate shot costs (up to $1.2 \cdot 10^6$), the shot-efficient CDR outperforms the standard algorithm. For example, efficient CDR provides improvement of the mitigated results over the noisy ones consistently for all $N_s^{\text{tot}} \geq 3 \cdot 10^3$. This should be compared with standard CDR that shows improvement only for much larger total budgets, $N_s^{\text{tot}} \geq 1.2 \cdot 10^5$. Thus, efficient CDR improves that shot budget threshold by almost two orders of magnitude. Similarly, the shot-efficient method obtains improvement by a factor of at least 4 with respect to the noisy results for all $N_s^{\text{tot}} \geq 7 \cdot 10^4$. The standard CDR method requires at least $N_s^{\text{tot}} = 1.2 \cdot 10^6$ to reach similar improvement.

4.2 LiH molecule ground state

4.2.1 The setup

As a second test case we consider error mitigation of the ground state energy of a LiH (lithium hydride) molecule at the equilibrium bond length 1.55 Å. An STO-3G basis set is used resulting in a 12-qubit Hamiltonian H_{LiH} which is a linear combination of $n_t = 631$ Pauli strings H^t

$$H_{\text{LiH}} = \sum_{t=1}^{n_t} c_t H^t, \quad (10)$$

with c_t being the coefficients. We derive the Hamiltonian using *Openfermion* package [65]. We approximate the ground state of the Hamiltonian with a hardware-efficient ansatz built out of 4 layers of general nearest-neighbor 2-qubit unitaries that were found by classical optimization of the ansatz. Subsequently, 2-qubit unitaries were compiled to native IBM gates [66] ($R_X(\pi/2) = e^{-i\pi/4}X$, $R_Z(\theta) = e^{-i\theta/2}Z$, CNOT) resulting in a circuit with 72 CNOTs, with up to 72 CNOTs in a causal cone of H^t , 12 non-parallelizable layers of CNOTs, and 369 non-Clifford $R_Z(\theta)$ gates.

4.2.2 Quantum computer noise model

We perform error mitigation using a noise model detailed in Appendix E that was derived using IBM's Ourense process tomography. We perform CDR for each term of the Hamiltonian H^t separately. As the expectation values of multiple terms of the Hamiltonian typically cannot be measured in the same computational basis, we estimate each $H_{\text{coi}}^{t,\text{noisy}}$ and $H_i^{t,\text{noisy}}$

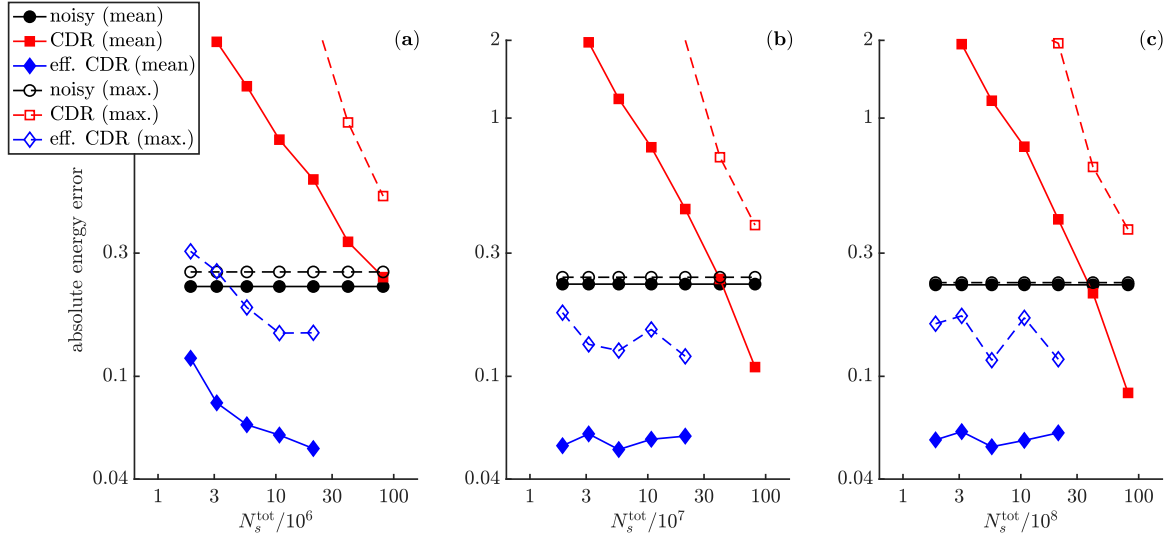


Figure 6: **Error mitigation of the ground state energy of LiH molecule.** Comparison of the standard and efficient CDR for the energy mitigation of the ground state of LiH molecule. Here, $Q = 12$, and we use IBM's Ourense-derived noise model from Appendix E. We plot the energy error obtained for $N_t \in \{2, \dots, 128\}$ ($N_t \in \{2, \dots, 32\}$) standard (efficient) CDR training circuits per Hamiltonian term. Furthermore, we consider $N_s = 10^3$ (a), $N_s = 10^4$ (b) and $N_s = 10^5$ shots (c) per H_t^{noisy} evaluation. For each (N_t, N_s) pair, we perform the standard (the red curves) and shot-efficient CDR (the blue curves) with 32 independent training sets and plot the mean (the solid lines) and maximal (the dashed lines) absolute energy errors (12) across the training sets versus the total shot cost N_s^{tot} (11). As a reference, we show the unmitigated energy errors as the black curves.

using independent shots. For each expectation value estimation, we use N_s shots. As the qubit Hamiltonian H_{LiH} does not have symmetries to exploit, for each H^t we construct efficient CDR training sets with $H_i^{t,\text{exact}} = -0.5 + (i-1)/(N_t-1)$, $i \in \{1, \dots, N_t\}$. We generate a different set of training circuits for each H_t . Here, N_t is a number of training circuits per Hamiltonian term. The total shot cost of the energy mitigation is

$$N_s^{\text{tot}} = n_t(N_t + 1)N_s. \quad (11)$$

The training circuits are generated by the optimization algorithm described in Appendix C.1, where we use $N = 10$. In this case we use the optimization approach because it is more time-efficient than the MCMC approach used for the XY model.

We compare the standard and efficient methods for the task of energy mitigation. We define

$$\text{absolute error} = \left| \sum_t c_t (H^{t,\text{noisy/mitigated}} - H^{t,\text{exact}}) \right|. \quad (12)$$

We evaluate the error for $N_t \in \{2, 4, 8, 16, 32\}$ and $N_s \in \{10^3, 10^4, 10^5\}$. For the standard method, we perform mitigation with even larger $N_t \in \{64, 128\}$. We perform error mitigation with 32 independent sets of the training circuits for each N_s and N_t pair. Each set of training circuits is obtained by randomly initializing the optimization procedure. For each (N_t, N_s) pair, we compute the average and the maximal error (12) across the training circuit sets and plot it versus N_s^{tot} in Figure 6.

We again find that the efficient method vastly outperforms the standard one. In particular, the efficient method converges quickly with increasing N_s^{tot} , unlike the standard one. Moderate $N_s^{\text{tot}} = 10^7$ is large enough to obtain converged energy error. In contrast, the standard CDR converges slowly and underperforms the efficient one even for the highest considered $N_s^{\text{tot}} = 10^{10}$. For moderate $N_s^{\text{tot}} = 10^6$ to $N_s^{\text{tot}} = 10^7$, the efficient mitigation error is an order of magnitude smaller than the standard one. For those N_s^{tot} and $N_s = 10^4, 10^5$, the efficient approach improves by a factor of 3.7 to 4.3 over the noisy results. Moreover, in Appendix H, we show that the performance of the efficient method does not depend significantly on the choice of N .

5 Conclusions and discussion

We proposed methods to improve the shot-efficiency of CDR. Our new methods rely on the observation that for large and deep circuits, the expectation values of near-Clifford circuits used by CDR to learn the noise effects cluster around 0. Because of that effect, a large number of shots is required to distinguish noisy expectation values. Consequently, a large number of shots are necessary to learn the correction successfully.

We proposed two strategies to alleviate this effect. The first one is to generate a distribution of near-Clifford training circuits with widely distributed exact expectation values for the circuit of interest. We

show that such a distribution can be efficiently generated with classical optimization or a semi-analytic procedure without requiring any quantum resources.

The second strategy is to exploit symmetries of the mitigated system by identifying multiple observables which by virtue of the symmetry have the same expectation values and can be measured simultaneously. In the presence of such symmetries, we propose that one performs CDR correction simultaneously for all observables by enforcing the symmetry constraints. This procedure effectively increases the number of available data points to optimize the CDR ansatz. It also restores symmetries of the error mitigated expectation values of the observables.

We have demonstrated that both strategies can be used to improve CDR shot-efficiency. We combined the strategies and applied them for error mitigation of the ground state of the XY Hamiltonian. This model displays translational and U(1) symmetries. The latter one is responsible for Hamming weight preservation. Using IBM Toronto we have demonstrated an order-of-magnitude improvement in shot-efficiency relative to standard CDR for moderate total shot costs ($3 \cdot 10^4$ to $3 \cdot 10^5$). Furthermore, we analyzed the performance of our shot-efficient method in more detail, using an IBM Ourense-based noise model, finding consistent improvement of our new method over standard methods for a large range of total error mitigation shot costs ($3 \cdot 10^3$ to 10^6). As a second example, we performed LiH ground state energy mitigation. For this task, we demonstrated three orders of magnitude reduction of the shot cost using the IBM Ourense-derived noise model despite the mitigated circuit not having exploitable symmetries.

The clustering problem affects all error mitigation methods that learn from near-Clifford circuits sharing Clifford gate arrangement with a general circuit of interest, including CDR generalizations with multiple noise levels in the training data [46, 47]. We expect that our proposed strategies will improve the shot-efficiency of those methods as well. The above techniques have been shown to require more shots than CDR to obtain substantial improvement over CDR. Therefore, improvements of their shot efficiency will be crucial for their practical application. Additionally, an open question is to what extent shot-efficiency of learning-based error mitigation can be improved by combining learning from shot-efficient sets of near-Clifford circuits with machine learning models capable of learning correction for multiple observables and circuits of interest [44]. Another practically relevant issue that needs further investigation is to what extent the shot-efficient methods' performance can be boosted by combination with specialized measurement error mitigation [67–69] or randomized compiling techniques [42].

Our numerical experiments and analytic arguments demonstrate that for a wide variety of problems, the

training circuits can be generated using classical resources that scale polynomially with the number of qubits and the number of gates in the circuit of interest. Nevertheless, an analysis of the computational time scaling is recommended while applying the methods to new circuit structures.

Moreover, it will be interesting to study the extent to which the proposed techniques affect the performance of CDR in the limit of large shot number. In particular, the symmetric CDR algorithm, apart from improving the shot-efficiency, restores symmetries of the mitigated system restricting possible error mitigation outcomes. A question arises whether such a restriction can improve the quality of error mitigation for degrees of freedom not fixed by the symmetry. Similarly, the usage of distributed training circuits in the limit of large shot number requires further examination. It has been observed that in some cases the use of training circuits with mitigated observable distribution close to the expectation value of the circuit of interest improves the quality of CDR corrections [38] in comparison to the standard Clifford method. It remains to be seen if this effect occurs for the training circuit distributions proposed here.

Finally, an interesting question is to what extent an application of the shot-efficient method can improve the quality of training for Variational Quantum Algorithms (VQAs). It has been shown that CDR can improve the trainability of VQAs by reversing distortions of the cost function landscape due to the noise [49]. At the same time, it has been shown that for limited shot resources, error mitigation can degrade VQAs performance by increasing the uncertainty of the cost function estimation [49]. We expect that by reducing the shot cost of the error mitigation up to orders of magnitude, the proposed method will enable an investigation of those effects with real-world hardware.

6 Code availability

Further implementation details are available from the authors upon request.

Acknowledgments

We thank Mike Martin for insightful discussions. PC acknowledges initial support from Laboratory Directed Research and Development (LDRD) program of Los Alamos National Laboratory (LANL) under project numbers 20190659PRD4 with subsequent support by the National Science Centre (NCN), Poland under project 2019/35/B/ST3/01028. MM and ATS acknowledge initial support from the Los Alamos National Laboratory (LANL) ASC Beyond Moore's Law project with subsequent support by the Laboratory Directed Research and Devel-

opment (LDRD) program of Los Alamos National Laboratory under project numbers 20210116DR and 20240712ER. This material is based upon work supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Quantum Science Center (LC). LC was also initially supported by the U.S. DOE, Office of Science, Office of Advanced Scientific Computing Research, under the Quantum Computing Application Teams program. This research used resources provided by the Los Alamos National Laboratory Institutional Computing Program, which is supported by the U.S. Department of Energy National Nuclear Security Administration under Contract No. 89233218CNA000001.

A Training circuit generation - standard algorithm.

Here we outline an algorithm to generate CDR training circuits used as a reference in the main text. The algorithm is based on random substitution of non-Clifford gates in a circuit of interest by nearby Clifford gates. The algorithm was originally proposed in [38] and was used and incorporated in subsequent error mitigation frameworks [47, 49]. Here we present it for a circuit compiled with native IBM gates X , $\sqrt{X}$, $R_Z(\theta)$ and CNOT.

X , $\sqrt{X}$ and CNOT are Clifford gates while $R_Z(\theta) = e^{-i\theta/2Z}$ is a Clifford gate only for $\theta = k\pi/2$, where k is an integer. Therefore, for a circuit containing $\tilde{N}$ non-Clifford R_Z gates we replace $\tilde{N} - N$ R_Z gates by Clifford R_Z rotations to obtain a training circuit with N non-Clifford gates. We denote angles of the non-Clifford rotations by $\theta_i, i \in \{1, \dots, \tilde{N}\}$. We replace these gates using the following procedure:

1. Take the circuit of interest as an input circuit. Repeat steps 2 and 3 $\tilde{N} - N$ times.
2. For each non-Clifford gate $R_Z(\theta_i)$ in the input circuit, calculate weights $w_{ik} = e^{-d_{ik}^2/\sigma^2}$, $k \in \{0, 1, 2, 3\}$. Here, $d_{ik} = \|e^{i\theta_i/2}R_Z(\theta_i) - e^{ik\pi/4}R_Z(k\pi/2)\|_F$, $\|\cdot\|_F$ is the Frobenius matrix norm and σ is a parameter. Note that d_{ik} quantifies the distance between $R_Z(\theta_i)$ and $R_Z(k\pi/2)$ in a way which does not depend on any global phase introduced by the action of these gates. We choose $\sigma = 0.5$ following [38, 47, 49].
3. Replace a gate $R_Z(\theta_i)$ by $R_Z(k\pi/2)$ according to the probability distribution

$$p_{ik} = \frac{w_{ik}}{\sum_{i=1}^m \sum_{k=0}^3 w_{ik}}, \quad (13)$$

where m is the number of non-Clifford gates in the input circuit, generating an output circuit. If it contains more than N non-Clifford gates, take

it as an input circuit for the next repetition of step 2.

4. Take the final output circuit as the training circuit.

B Existence of well-distributed $N = 1$ training circuits

Here, we consider a circuit of interest compiled to CNOTs and general single-qubit unitaries of form

$$u(\theta_1, \theta_2, \theta_3) = R_Z(\theta_1)\sqrt{X}R_Z(\theta_2)\sqrt{X}R_Z(\theta_3), \quad (14)$$

with $\sqrt{X} = R_X(\pi/2) = e^{-i\pi/4X}$ up to a global phase, and $R_Z(\theta) = e^{-i\theta/2Z}$. We note that CNOTs, R_Z , and $\sqrt{X}$ are native gates of IBM's quantum computers. We consider an observable of interest being a Pauli string $P = \bigotimes_{i=1}^Q P_i$, with, $P_i \in \{X, Y, Z, I\}$ and I being an identity. In this Appendix, we analyze a distribution of expectation values of near-Clifford CDR training circuits with $N = 1$ non-Clifford R_Z gates for such a circuit of interest.

We employ here Heisenberg picture in which a unitary U acts at an observable O as

$$O \xrightarrow{U} U^\dagger O U \quad (15)$$

This corresponds to the unitary acting on a state as

$$|\psi\rangle \xrightarrow{U} U|\psi\rangle. \quad (16)$$

We first note that a general Pauli string P can be transformed to a Pauli Z string, $P_Z = \bigotimes_{i=1}^Q P_i^Z$, with $P_i^Z \in \{Z, I\}$, by a shallow unitary built from single-qubit native Clifford gates, as we have,

$$Y \xrightarrow{R_Z(\pi)R_X(\pi/2)} Z, \quad (17)$$

$$X \xrightarrow{R_Z(\pi/2)R_X(\pi/2)} Z. \quad (18)$$

Therefore, by applying at the end of the circuit of interest $R_Z(\pi/2)\sqrt{X}$ ($R_Z(\pi)\sqrt{X}$) to qubits at which we have X (Y) in P , we transform P to P_Z with $P_i^Z = Z$ if $P_i \in \{X, Y, Z\}$, and $P_i^Z = I$ if $P_i = I$. This basis transformation reduces the case of an arbitrary Pauli string to the special case of Pauli Z string, while adding a small constant overhead in terms of the circuit depth. Furthermore, a total number of CNOTs in the circuit of interest is not changed.

To establish a general result, we first consider a toy example of $Q = 1$,

$$U = u(\theta_1, \theta_2, \theta_3), \quad (19)$$

with $\theta_1, \theta_2, \theta_3 \in [0, 2\pi)$, and $P^Z = Z$. In the Heisenberg picture, we have

$$Z \xrightarrow{R_Z(\theta_1)} Z, \quad (20)$$

$$Z \xrightarrow{R_X(\pi/2)} Y \quad (21)$$

$$Y \xrightarrow{R_Z(\theta_2)} \cos(\theta_2)Y + \sin(\theta_2)X, \quad (22)$$

$$\cos(\theta_2)Y + \sin(\theta_2)X \xrightarrow{R_X(\pi/2)} -\cos(\theta_2)Z + \sin(\theta_2)X, \quad (23)$$

$$\begin{aligned} & -\cos(\theta_2)Z + \sin(\theta_2)X \xrightarrow{R_Z(\theta_3)} \cos(\theta_2)Z + \\ & -\sin(\theta_2)(\cos(\theta_3)X - \sin(\theta_3)Y). \end{aligned} \quad (24)$$

Then,

$$\langle P^Z \rangle = \text{Tr} \left(|0\rangle\langle 0| (-\cos(\theta_2)Z + \sin(\theta_2)(\cos(\theta_3)X - \sin(\theta_3)Y)) \right), \text{ by } u(\pi, \pi, 0). \quad (25)$$

yielding

$$\langle P^Z \rangle = -\cos(\theta_2). \quad (26)$$

Therefore, we can set $R_Z(\theta_1)$ and $R_Z(\theta_3)$ to be Cliffords and obtain an arbitrary $-1 \leq \langle P^Z \rangle \leq 1$ by adjusting θ_2 . If $P^Z = I$ instead, $P^Z \xrightarrow{U} P^Z$, and $\langle P^Z \rangle = 1$.

Next, we note that we can easily generalize this result to the case of the observable of interest being an arbitrary Pauli Z string P_Z and U being $u(\theta_1, \theta_2, \theta_3)$ acting at a qubit j . If $P_j^Z = Z$, we obtain

$$P^Z \xrightarrow{U} -\cos(\theta_2)P^Z + \sin(\theta_2)\cos(\theta_3)\tilde{P} - \sin(\theta_2)\sin(\theta_3)\tilde{\tilde{P}}, \quad (27)$$

where

$$\tilde{P} = X_j \bigotimes_{i \neq j} P_i^Z, \quad (28)$$

$$\tilde{\tilde{P}} = Y_j \bigotimes_{i \neq j} P_i^Z, \quad (29)$$

and

$$\langle P^Z \rangle = \text{Tr}(U^\dagger P^Z U |0\rangle\langle 0|) = -\cos(\theta_2), \quad (30)$$

where $|0\rangle = \bigotimes_{i=1}^Q |0\rangle_i$. While, if $P_j^Z = I$, then

$$P^Z \xrightarrow{U} P^Z, \quad (31)$$

$$\langle P^Z \rangle = 1. \quad (32)$$

Furthermore, we note that a CNOT gate maps a Z string into a Z string as

$$Z_1 \xrightarrow{\text{CNOT}_{12}} Z_1, \quad (33)$$

$$Z_2 \xrightarrow{\text{CNOT}_{12}} Z_1 Z_2, \quad (34)$$

$$Z_1 Z_2 \xrightarrow{\text{CNOT}_{12}} Z_2. \quad (35)$$

Here, CNOT_{12} is controlled at qubit 1 and targets qubit 2. We also note that a CNOT acting on the initial state $|0\rangle$ does not change it, and that one can express I (up to a global phase) as $u(\theta_1, \theta_2, \theta_3)$ such that its native gate decomposition (14) is built from Clifford gates, e.g.,

$$I \propto u(\pi, \pi, 0). \quad (36)$$

Now, we propose an algorithm to find a near-Clifford training circuit with a single non-Clifford R_Z – for a circuit of interest compiled to u and CNOT gates and an observable of interest $P^Z \neq I$ – such that $\langle P^Z \rangle = -\cos(\theta_2)$, $\theta_2 \in [0, 2\pi)$.

1. Apply gates in the circuit of interest in the Heisenberg picture until u with a non-trivial dependence of $\langle P^Z \rangle$ on θ_2 (i.e. as in (30)) is found.
2. Construct the training circuit by replacing the found gate by e.g. $u(0, \theta_2, 0)$ and all other u gates

Note that for such a choice we can easily evaluate $\langle P^Z \rangle$ by combining Schrödinger and Heisenberg picture. Namely we divide the circuit into two parts. The first one is built from the gates preceding $u(0, \theta_2, 0)$ in the Heisenberg picture and $u(0, \theta_2, 0)$. It corresponds to a unitary U_1 . The second one is composed of the rest of the gates and corresponds to a unitary U_2 . We have

$$\langle P^Z \rangle = \text{Tr}(U_2 |0\rangle\langle 0| U_1^\dagger P^Z U_1), \quad (37)$$

but as the gates in U_2 do not change the initial state being CNOTs or compilations of identity, we obtain

$$\langle P^Z \rangle = \text{Tr}(|0\rangle\langle 0| U_1^\dagger P^Z U_1) = -\cos(\theta_2). \quad (38)$$

We note that in principle for some choices of P_Z and the circuit of interest a u gate with the non-trivial dependence cannot be found and the algorithm fails to produce the training circuit. Nevertheless, for a non-trivial $\langle P^Z \rangle$ that contains Z 's and for popular variational ansatzes the algorithm is guaranteed to produce the training circuit. Examples of such ansatzes are a hardware-efficient ansatz that is constructed from layers of CNOTs decorated by u gates covering all the qubits and a QAOA ansatz that has a mixer compiling to a product of u gates acting at all the qubits. The identification of the non-trivial gate can be simplified by simulating a Clifford circuit, which can be performed efficiently using classical computation. [70]. Finally, we note that usually multiple u gates affecting $\langle P_Z \rangle$ can be found. In such a case, one has a freedom to choose location of the near-Clifford gate or one can attempt to construct $N > 1$ training circuits with an arbitrary $\langle P_Z \rangle$, as demonstrated numerically in Appendix C.

C Shot-efficient training circuits generation

C.1 Near-Clifford circuits optimization.

In this section we present an algorithm for fast generation of near-Clifford training circuits. We start with a circuit of interest $\mathcal{C}$ and an observable O . The circuit $\mathcal{C}$ is used to prepare state $|\psi_{\mathcal{C}}\rangle$ on a noisy quantum

computer. Error mitigation schemes aim to correct $\langle\psi_C|O|\psi_C\rangle$ measured on that computer.

We assume that $\mathcal{C}$ is a brick layer circuit, in which every two-qubit gate has the form $u_j u_{j+1} \text{CNOT}_{j,j+1}$, where

$$u_j = R_Z(\theta_{j,1})\sqrt{X}R_Z(\theta_{j,2})\sqrt{X}R_Z(\theta_{j,3}). \quad (39)$$

Here, $\sqrt{X} = e^{-i\pi/4X}$ and $R_Z(\theta) = e^{-i\theta/2Z}$. We also assume that $O = Z_k$, where k is a qubit located in the middle of the circuit. The method presented below can be easily extended to more general circuits and other observables.

The CDR method considered in the main text constructs a collection of circuits $\{\mathcal{C}_i\}$ with the same structure as $\mathcal{C}$ such that $\langle\psi_{\mathcal{C}_i}|O|\psi_{\mathcal{C}_i}\rangle = y_i$. Here, $y_i \in [-1, 1]$ are preselected, conveniently distributed values. Circuits $\mathcal{C}_i$ are built in the same way as $\mathcal{C}$. The arrangement of all the gates in $\mathcal{C}_i$ is the same as in $\mathcal{C}$. The circuit differs only by the angles θ .

The method consists of two steps: (i) project $\mathcal{C}$ to a near-Clifford circuit $\mathcal{C}_i(\theta)$ and (ii) find θ such that

$$|\langle\psi_{\mathcal{C}_i(\theta)}|O|\psi_{\mathcal{C}_i(\theta)}\rangle - y_i| < \epsilon \quad (40)$$

for a desired precision ϵ .

In step (i), a small number of one-qubit gates u in $\mathcal{C}$ is randomly selected. That set of gates is then expanded by one-qubit gate that acts on qubit k just before the observable $O = Z_k$ is measured on that qubit. Those gates are replaced according to Eq. (39) with random angles $\theta \in (0, 2\pi)$. All the other one-qubit gates are replaced also according to Eq. (39) but with angles θ chosen from a discrete sets: $\theta_1, \theta_3 \in \{0, \pi/2, \pi, 3\pi/2\}$ and $\theta_2 \in \{0, \pi\}$. This way, $\mathcal{C}_i(\theta)$ is a near-Clifford circuit with single, general one-qubit gate placed at the end of the circuit, at a qubit that is later measured. Appendix B guarantees that there exists θ such that the condition in Eq. (40) is met.

In step (ii), the optimization

$$\min_{\theta} |\langle\psi_{\mathcal{C}_i(\theta)}|O|\psi_{\mathcal{C}_i(\theta)}\rangle - y_i| \quad (41)$$

is performed until the value of the optimized function is below predefined ϵ .

The efficiency of the approach described here relies on the optimization performed in step (ii). The existence of a solution is guaranteed but an estimate of the time that is spent on finding parameters θ in (41) may be difficult to obtain. We stress that even though parameters θ can be found by following the procedure outlined in Appendix B, the error mitigation method needs a diverse set of circuits to work accurately and training circuits should contain more than one non-Clifford gate. In that setting, the numerical optimization over θ becomes necessary. Here, we perform numerical simulations to estimate the time needed to derive a single near-Clifford circuit $\mathcal{C}_n$.

The circuit of interest $\mathcal{C}$ is chosen randomly as described above, with one-qubit gates selected according to Eq. (39). We consider circuits of sizes up to

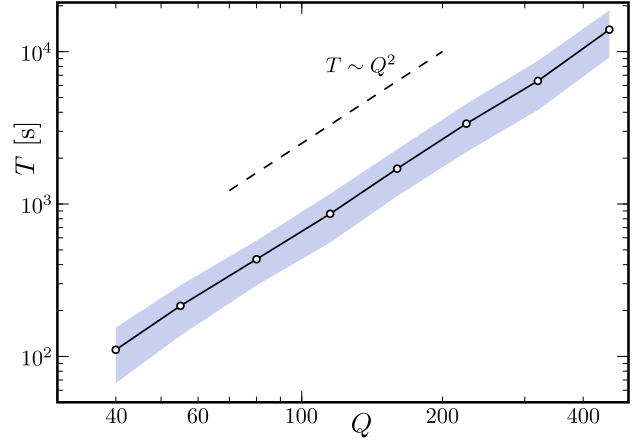


Figure 7: Computational cost of obtaining the training data. The plot shows the scaling of the time T to find one near-Clifford $N = 10$ circuit $\mathcal{C}_i$ with the number of qubits Q . We start from a random circuit whose depth L is linear in Q . We find that the computation time T scales as $\sim Q^2$. This scaling closely matches the time to simulate near-Clifford circuit, which indicates that the number of cost function (41) evaluations to find optimal θ does not itself scale with Q . This leads to an efficient generation of the training data.

$Q = 455$ qubits. The circuit depth L has been grown linearly with the number of qubits, $L = Q/5$. The largest circuit $\mathcal{C}$ that we analyze here contains over 41,000 CNOT gates and 83,000 one-qubit gates. We are considering an observable $O = Z_{\lfloor Q/2 \rfloor}$ and work with $\epsilon = 10^{-3}$ and $N = 10$ non-Clifford gates. We use off-the-shelf optimization methods to optimize over angles θ .

Figure 7 shows time T to find circuit $\mathcal{C}_i$ as a function of the number of qubits Q . The value of y_i in (41) is chosen randomly from $[-1, 1]$ for each $\mathcal{C}_i$. The time is averaged over 200 random circuits $\mathcal{C}$. The optimization is executed on a single core. The solid line shows average time and gray area represents variance in the data. The experimentally obtained time to find solution T scales roughly as Q^2 , as indicated by dashed line on the plot. This dependence comes from simulation of near-Clifford circuits which scales as the number of gates in the circuit. In our case, that number scales as Q^2 . Our numerical results thus show that there is no overhead from optimization even in the limit of large system sizes and deep circuits. The simple optimization method used here is sufficient to reliably and quickly find near-Clifford circuits needed for the error mitigation method considered in the main text.

C.2 Markov Chain Monte Carlo near-Clifford circuit sampling.

Here we present an alternative algorithm to numerically generate a distribution of training circuits with widely distributed values of O^{exact} used by the shot-

efficient CDR method. The algorithm attempts to generate training circuits with values of O^{exact} close to target values $y_1, y_2, \dots, y_n$ using Markov Chain Monte Carlo (MCMC) sampling [71] of near-Clifford circuits. For each target value, y_k , we generate an MCMC chain built from near-Clifford circuits with N non-Clifford gates using a target probability distribution $p \propto e^{-(O^{\text{exact}} - y_i)^2 / \sigma_{\text{MCMC}}^2}$ where O^{exact} is the exact expectation value of the mitigated observable for a near-Clifford circuit and σ_{MCMC} is a constant. We select a circuit with O^{exact} close to y_i as a training circuit. In this work, O is a Pauli string and we choose $y_i = -0.5 + \frac{i-1}{n-1}$, $i \in \{1, 2, \dots, n\}$. For this choice, we have a distribution of target values $|y_n - y_1| = 1$ which is comparable to the maximal possible spread of O^{exact} (i.e. 2). Furthermore, we choose $\sigma_{\text{MCMC}} = 0.01$.

We use a version of the Metropolis-Hastings algorithm [71] to generate an MCMC chain for a given y_i . We assume here that near-Clifford circuits in the chain can be obtained by replacing most of the non-Clifford gates in the circuit of interest by Clifford gates and consequently the circuit of interest can be obtained by replacing some of the Clifford gates in circuits from the chain by non-Clifford gates. The algorithm is:

1. Initialize the chain taking as its first element a near-Clifford circuit with N non-Clifford gates obtained by substituting most of non-Clifford gates in the circuit of interest by Clifford gates. In our implementation, we use an algorithm from Appendix A to generate the first circuit. Repeat steps 2 and 3 until a circuit with O^{exact} close enough to y_i is generated.
2. Prepare a candidate circuit by taking the last circuit of the chain and replacing N_g randomly chosen non-Clifford gates by Clifford gates and N_g randomly chosen Clifford gates by the corresponding non-Clifford gates of the circuit of interest. Note that the new circuit has N non-Clifford gates which are the same as corresponding non-Clifford gates in the circuit of interest.
3. Compute the ratio $w_{\text{cand}}/w_{\text{accept}}$, where

$$w_{\text{cand}} = e^{-(y_{\text{cand}} - y_i)^2 / \sigma_{\text{MCMC}}^2}, \quad (42)$$

$$w_{\text{accept}} = e^{-(y_{\text{accept}} - y_i)^2 / \sigma_{\text{MCMC}}^2} \quad (43)$$

and y_{cand} , y_{accept} are O^{exact} for the new circuit and the last circuit of the chain, respectively. Generate a random number $u \in [0, 1]$. If $u < w_{\text{cand}}/w_{\text{accept}}$, accept the new circuit as the last circuit of the chain. Otherwise add a copy of the last circuit to the chain.

In the limit of long MCMC chains this algorithm is guaranteed to generate a distribution of near-Clifford circuits converging to the target distribution $p \propto$

$e^{-(O^{\text{exact}} - y_i)^2 / \sigma_{\text{MCMC}}^2}$ [71] if near-Clifford circuits with O^{exact} close enough to y_i exist. The existence of such circuits for a wide class of ansatzes is guaranteed by Appendix B.

We implement step 2 setting $N_g = 5$ and replace $R_Z(\gamma)$ by $R_Z(k\pi/2)$ according to a probability distribution

$$p_k = \frac{w_k}{\sum_{k=0}^3 w_k}, \quad w_k = e^{-d_k^2 / \sigma^2}, \quad (44)$$

$$d_k = \|e^{i\theta/2} R_Z(\theta) - e^{ik\pi/4} R_Z(k\pi/2)\|_{\text{F}}.$$

Here, we set $\sigma = 0.5$. We find that such implementation results in fast convergence of the MCMC chains to the target distribution.

D Implementing symmetry constraints

For a system of M observables, we implement the constraint (6) by defining a penalty function:

```
def condition(x):
    return std(x[:M] * 0_coi + x[M:])**2

@linear_inequality(condition)
def penalty(x):
    return 0.0
```

where $\mathbf{x}[:M]$ are the coefficients $a_1, \dots, a_M$, $\mathbf{x}[M:]$ are the coefficients $b_1, \dots, b_M$, and 0_coi is an array of noisy expectation values.

Here, we use `std` from `numpy` for an array-based calculation of standard deviation, and `linear_inequality` from `mystic.penalty` to create a penalty function that scales linearly when `condition` is greater than zero.

To ensure that the `condition` is not violated during the minimization of the cost function (5), we convert the `penalty` to a `constraint` operator:

```
kwds = dict(nvars=2*M, termination=stop)
constraint = as_constraint(penalty, **kwds)
```

using `as_constraint` from `mystic.constraints`, where `stop` is the optimization termination condition defined by using `ChangeOverGeneration` from `mystic.termination`:

```
args = dict(tolerance=1e-12, generations=60)
stop = ChangeOverGeneration(**args).
```

The operator `constraint` is then applied during minimization of the cost function as $\mathbf{x}' = \text{constraint}(\mathbf{x})$, to ensure all candidate solutions $\mathbf{x}'$ for coefficients a and b generated by the optimizer satisfy the constraints. This is done by passing `constraint` to the `constraints` keyword of the `mystic` optimizer used to minimize the cost function.

E The noise model

We construct the noise model used in Sections 4.1.3, 4.2.2 with process matrices obtained by gate set tomography (GST) experiments performed on the IBM Ourensense quantum computer. These matrices are given in Appendix B of [72], where the details of the experiments can be found. Our goal in Section 4.1.3 is to investigate shot-efficiency gains for deep circuits that are challenging to mitigate with current hardware. To achieve that, we reduce the effective noise rates of the model in comparison to the original process matrices' noise rates. Such a reduction is further justified by the fact that IBM Ourensense's noise rates are higher than the error rates of current IBM quantum computers. Furthermore, in our noise model we assume that the noise rates of gates vary for different qubits and different connections between qubits as in the case of real devices.

More explicitly, in our noise model we assume (following [72]) that native gates of the device are $\sqrt{X} = e^{-i\pi/4X}$, CNOT, identity I and $R_Z(\theta) = e^{-i\theta/2Z}$. R_Z is assumed to be perfect for IBM quantum computers, as it can be performed "virtually" in a zero-duration fashion without incurring significant error [52]. We construct process matrices of I and $\sqrt{X}$ as convex combinations of the processes matrices from [72] $\sqrt{X}^{\text{Ourensense}}, I^{\text{Ourensense}}, \text{CNOT}^{\text{Ourensense}}$ and the noiseless ones $\sqrt{X}^{\text{perfect}}, I^{\text{perfect}}, \text{CNOT}^{\text{perfect}}$

$$\sqrt{X}^i = p_i \sqrt{X}^{\text{Ourensense}} + (1 - p_i) \sqrt{X}^{\text{perfect}}, \quad (45)$$

$$I^i = p_i I^{\text{Ourensense}} + (1 - p_i) I^{\text{perfect}}, \quad (46)$$

$$\text{CNOT}^{ij} = p_{ij} \text{CNOT}^{\text{Ourensense}} + (1 - p_{ij}) \text{CNOT}^{\text{perfect}}. \quad (47)$$

Here, indices i and j number qubits and p_i, p_{ij} were drawn randomly from a uniform probability distribution on the interval $[0.05, 0.15]$.

To quantify quality of a noisy gate g in our model we use average process infidelity defined as

$$1 - F(g^{\text{noisy}}, g^{\text{perfect}}) = 1 - \int d\psi \text{Tr}[g^{\text{noisy}}(|\psi\rangle\langle\psi|)g^{\text{perfect}}(|\psi\rangle\langle\psi|)], \quad (48)$$

where $g^{\text{noisy}}(|\psi\rangle\langle\psi|)$ and $g^{\text{perfect}}(|\psi\rangle\langle\psi|)$ are states obtained by action of the noisy and the noiseless gates at a pure state $|\psi\rangle\langle\psi|$, respectively. The integral for single-qubit (two-qubit) gates is taken over single-qubit (two-qubit) Haar measure. The Ourensense process matrices have infidelities [72]

$$1 - F(I^{\text{Ourensense}}, I^{\text{perfect}}) = 2.8 \cdot 10^{-3}, \quad (49)$$

$$1 - F(\sqrt{X}^{\text{Ourensense}}, \sqrt{X}^{\text{perfect}}) = 8.8 \cdot 10^{-4}, \quad (50)$$

$$1 - F(\text{CNOT}^{\text{Ourensense}}, \text{CNOT}^{\text{perfect}}) = 1.9 \cdot 10^{-2}. \quad (51)$$

As the process matrices used (45, 46, 47) are convex combinations of the Ourensense and noiseless process matrices and fidelity (48) is linear in the left argument, we have:

$$1 - F(g^i, g^{\text{perfect}}) = p_i(1 - F(g^{\text{Ourensense}}, g^{\text{perfect}})), \quad (52)$$

$$1 - F(\text{CNOT}^{ij}, \text{CNOT}^{\text{perfect}}) = p_{ij}(1 - F(\text{CNOT}^{\text{Ourensense}}, \text{CNOT}^{\text{perfect}})), \quad (53)$$

with g being I or $\sqrt{X}$. Consequently, the model gates infidelities are on average reduced with respect to IBM's Ourensense device by an order of magnitude.

Furthermore, we assume the presence of state preparation error. We assume that the initial state ρ_0 is a product state of single-qubit states

$$\rho_0 = \rho_0^1 \otimes \cdots \otimes \rho_0^N \quad (54)$$

with

$$\rho_0^i = p_i \rho_0^{\text{Ourensense}} + (1 - p_i) \rho_0^{\text{perfect}}, \quad (55)$$

where $\rho_0^{\text{Ourensense}}$ corresponds to the initial state obtained by GST experiments for IBM Ourensense and given in Appendix B of [72], while $\rho_0^{\text{perfect}} = |0\rangle\langle 0|$ corresponds to perfect state preparation. In our noise model we neglect measurement error as it can be efficiently mitigated by specialized techniques [67–69]. Probabilities p_i were drawn randomly from a uniform probability distribution on the interval $[0.05, 0.15]$.

To quantify the noisy state preparation quality we use state infidelity

$$1 - F(\sigma, \rho) = 1 - (\text{Tr} \sqrt{\sqrt{\rho} \sigma \sqrt{\rho}})^2. \quad (56)$$

For the Ourensense state preparation we have [72]

$$1 - F(\rho_0^{\text{Ourensense}}, \rho_0^{\text{perfect}}) = 9.7 \cdot 10^{-3}, \quad (57)$$

while for the noise model we obtain

$$1 - F(\rho_0^i, \rho_0^{\text{perfect}}) = 9.7 \cdot 10^{-3} p_i, \quad (58)$$

i.e. the model infidelity is on average reduced with respect to the real device by an order of magnitude.

F Training sets from the real-hardware implementation

For the standard CDR we generate training circuits by random substitutions as described in Appendix A. We measure the circuit of interest in one basis to determine expectation values of $O_i = X_i X_{i+Q/2}, i \in \{1, \dots, Q/2\}$ correlators and in another one to determine expectation values of $O_{i+Q/2} = Y_i Y_{i+Q/2}, i \in \{1, \dots, Q/2\}$ correlators. We consider cases of $N_t \in \{2, 6, 8, 12, 30\}$ training circuits to correct O_i^{noisy} . To gather statistics for each case we generate 10 independent sets of training circuits. As we need to measure

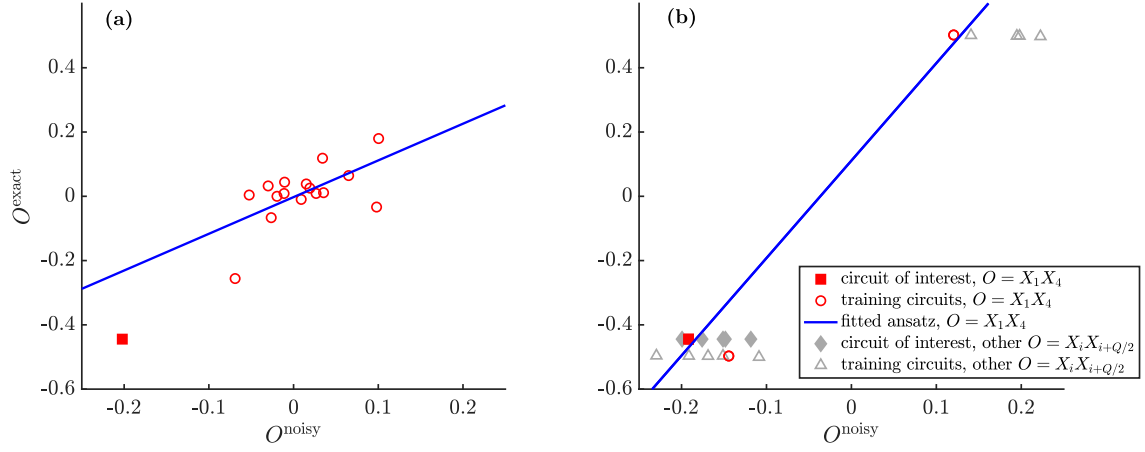


Figure 8: Training sets from the real-hardware implementation. In (a), we show a standard CDR training set for the real-hardware implementation from Figure 4 and the ground state expectation value of $X_1 X_4$ (red square). Here we use 15 training circuits (red circles) to correct $X_1 X_4$. Number of shots per expectation value evaluation is $N_s = 10^4$. Due to clustering of the training circuits quality of the CDR mitigation is poor as shown by the fitted CDR ansatz (blue line). We obtained similar results for other correlators (8). In (b), we show a training set for the shot-efficient method and $N_t = 12$. Here we correct all symmetric half-chain correlators (8) using the symmetric CDR algorithm and generate a widely-spread distribution of the training circuits for each observable independently. Red squares and circles show expectation values of $X_1 X_4$ for the training data and the circuit of interest. Gray squares and circles show expectation values of the other symmetric correlators. As in (a), we have $N_s = 10^4$. We obtain good quality of the error mitigation despite total shot cost of the error mitigation N_s^{tot} being smaller than in (a). For the sake of transparency we show a CDR fit only for $X_1 X_4$. For other correlators (8), quality of the error mitigation is similar.

$X_i X_{i+Q/2}$ and $Y_i Y_{i+Q/2}$ independently, for $N_t > 2$ we measure randomly chosen $N_t/2$ circuits in the $X_i X_{i+Q/2}$ basis and another half of the training circuits in the $Y_i Y_{i+Q/2}$ basis. This strategy cannot be used for $N_t = 2$, so in that case for half of the sets of the training circuits we measure $X_i X_{i+Q/2}$ correlators and for the other half $Y_i Y_{i+Q/2}$ correlators. To randomize the results further, while performing CDR correction for each set of the training circuits, we measure the circuit of interest independently. Figure 8(a) shows a typical (resulting in median value of the mitigated error (9)) training set generated by that procedure for $O_1 = X_1 X_4$ and $N_t = 30$, $N_s = 10^4$. Obtained training sets are strongly affected by clustering of the training circuits in the same way as in the case of training sets generated with the noise model in Figure 2. That explains poor performance of the standard CDR.

In the case of the shot-efficient version we generate distributions of widely-distributed training circuits for each O_i independently. To that end we use an algorithm from Appendix C.2. In the case of $N_t \in \{12, 18, 30\}$ for each O_i we generate training circuits with values of $O_i^{\text{exact}} \approx -0.5, 0.5$, $O_i^{\text{exact}} \approx -0.5, 0, 0.5$, and $O_i^{\text{exact}} \approx -0.5, -0.25, 0, 0.25, 0.5$, respectively. For $N_t < 12$ we choose randomly $N_t/2$ observables O_i for which we generate the training circuits with $O_i^{\text{exact}} \approx -0.5, 0.5$. To randomize results for each N_t we consider 10 different sets of training circuits obtained by starting MCMC chains with different initial circuits. Additionally, for each set of training circuits we use independent measurements of

the circuit of interest. We show a typical training set for $N_t = 12$ in Figure 8(b).

G Effects of hardware drift on performance of learning-based error mitigation

Shot-efficiency of the error mitigation is crucial for obtaining good quality of quantum computation when shot resources are limited. Here we argue that shot resources are not only limited by available time of a quantum computer. They are also effectively limited by its variability in time. In the case of the learning-based error mitigation we learn the correction from training circuits assuming that the noise remains constant in time. That assumption is an idealization of real-world devices. In reality, certain characteristics of such devices “drift” in time. The shot-efficient method proposed here enables to perform the error mitigation over a shorter period of time than the standard method. Therefore, it violates the constant noise assumption less severely than the standard CDR method. Consequently, we expect it to deliver better quality of the error mitigation than the standard approach. While we leave quantifying this effect to a future work, we show a proof of principle results demonstrating that the drift affects the quality of the error mitigation. Namely, we performed shot-efficient CDR error mitigation for the symmetric half-chain correlators (8) of the ground state of the 6-qubit XY model using IBM Toronto at two different times. In both cases we used the same training circuits which

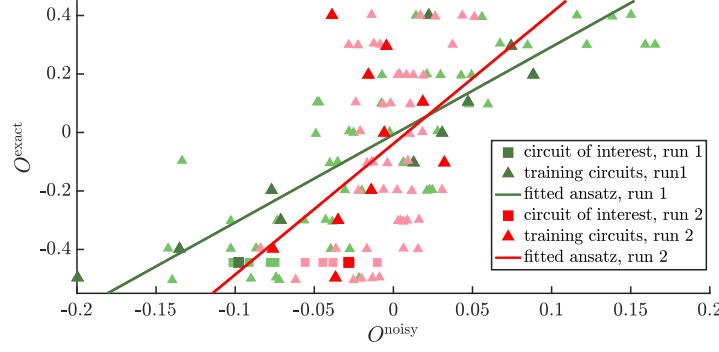


Figure 9: **Effects of the hardware drift.** We compare two shot-efficient CDR runs conducted for the same circuit of interest at two different times. We mitigate the half-chain correlators (8) of the ground state of the 6-qubit XY model using IBM Toronto. Run number 1 was conducted on 01/22/2022 while run number 2 was performed on 03/23/2022. In both runs we used the same training circuits which were compiled to the native gates in the same way. We show here training sets and fitted ansatzes from both runs. Green squares and triangles are the results from the run number 1 while the red ones are results from run 2. We use dark red and dark green symbols to show X_1X_4 and light red and light green ones to plot the other correlators (8). The drift changed distribution of O^{noisy} affecting parameters of the fitted ansatzes and quality of CDR mitigation. Here, for the sake of transparency we plot only CDR fits (the lines) for X_1X_4 . The fits for other correlators give similar quality of the correction as the ones shown here. For the run 1 we obtain the mitigated error 0.14 while for the run 2 the mitigated error is 0.28. We remark that here quality of the error mitigation is in both cases worse than the one reported in Figures 4, 8 as we use 6 of 27 qubits of the device with larger error rates than the ones used in Figures 4, 8.

we compiled to the native gates in the same way. We found that coefficients of the fitted CDR ansatzes and average quality of the error mitigation are different for those two runs, see Figure 9. For the run number 1 which occurred on 03/23/2022 we obtained the mitigated error 0.14. For the run number 2 which was conducted on 01/22/2022 we obtained the mitigated error 0.28. Mitigated errors are computed according to (9). That comparison shows that, indeed, the device drift can significantly alter results of the error mitigation.

H Impact of the number of non-Clifford gates in the training circuits, N , on the efficiency of CDR performance.

Here, we examine the effects of a choice of a CDR hyperparameter N on the performance of the shot-efficient method for the LiH ground state error mitigation from Section 4.2. We choose the same circuit of interest, observable of interest, noise model, and training circuit construction method as in Section 4.2. For such a choice, we compute the absolute mitigated energy error (12) for $N \in \{1, 5, 10, 20\}$, as shown in Figure 10. This comparison is performed with $N_t = 2$ and $N_s \in \{10^3, 10^4, 10^5\}$. We find that the error remains approximately constant as a function of N .

References

- [1] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, et al. “Quantum supremacy using a programmable superconducting processor”. *Nature* **574**, 505–510 (2019).
- [2] Han-Sen Zhong, Yu-Hao Deng, Jian Qin, Hui Wang, et al. “Phase-programmable gaussian boson sampling using stimulated squeezed light”. *Phys. Rev. Lett.* **127**, 180502 (2021).
- [3] Samson Wang, Enrico Fontana, M. Cerezo, Kunal Sharma, Akira Sone, Lukasz Cincio, and Patrick J Coles. “Noise-induced barren plateaus in variational quantum algorithms”. *Nature Communications* **12**, 1–11 (2021).
- [4] Daniel Stilck França and Raul Garcia-Patron. “Limitations of optimization algorithms on noisy quantum devices”. *Nature Physics* **17**, 1221–1227 (2021).
- [5] Sitan Chen, Jordan Cotler, Hsin-Yuan Huang, and Jerry Li. “Exponential separations between learning with and without quantum memory”. In 2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS). Pages 574–585. IEEE (2022).
- [6] Abhinav Kandala, Kristan Temme, Antonio D. Córcoles, Antonio Mezzacapo, Jerry M. Chow, and Jay M. Gambetta. “Error mitigation extends the computational reach of a noisy quantum processor”. *Nature* **567**, 491–495 (2019).
- [7] Ryan LaRose, Andrea Mari, Vincent Russo, Dan Strano, and William J. Zeng. “Error mitigation increases the effective quantum volume of quan-

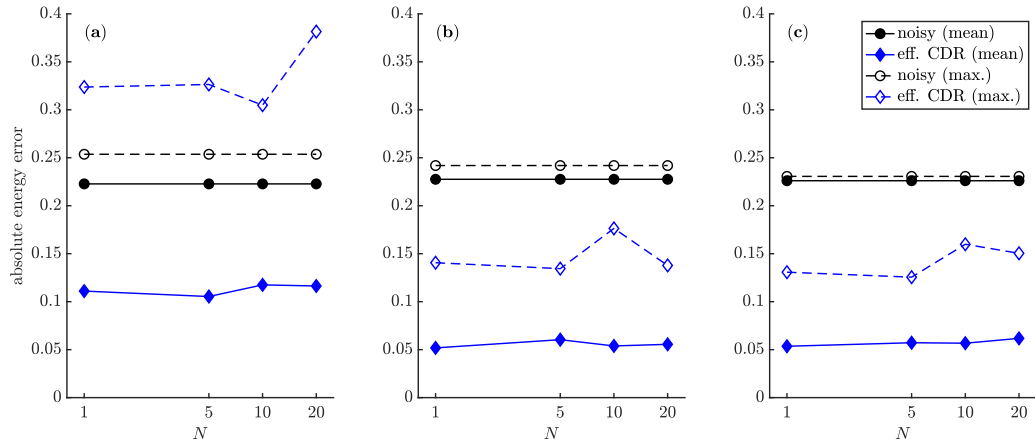


Figure 10: **Error mitigation of the ground state energy of LiH molecule for different choices of N .** The mean and maximal absolute energy errors (12) of the efficient CDR method for different choices of a CDR hyperparameter $N \in \{1, 5, 10, 20\}$. Here, we mitigate the ground state of LiH molecule using IBM’s Ourense-derived noise model. The circuit of interest, the noise model, and the training circuit construction method are the same as in Sec. 4.2. Here, $N_t = 2$, and we show $N_s = 10^3, 10^4, 10^5$ results in panels (a), (b), (c), respectively. The mean and maximal values are obtained using a sample of 32 independent sets of training circuits for each N value. As a reference, we also show errors of the noisy unmitigated data.

- tum computers” (2022). url: <https://arxiv.org/abs/2203.05489>.
- [8] Suguru Endo, Zhenyu Cai, Simon C Benjamin, and Xiao Yuan. “Hybrid quantum-classical algorithms and quantum error mitigation”. *Journal of the Physical Society of Japan* **90**, 032001 (2021).
 - [9] Kristan Temme, Sergey Bravyi, and Jay M. Gambetta. “Error mitigation for short-depth quantum circuits”. *Physical review letters* **119**, 180509 (2017).
 - [10] Eugene F Dumitrescu, Alex J McCaskey, Gaute Hagen, Gustav R Jansen, Titus D Morris, T Papenbrock, Raphael C Pooser, David Jarvis Dean, and Pavel Lougovski. “Cloud quantum computing of an atomic nucleus”. *Phys. Rev. Lett.* **120**, 210501 (2018).
 - [11] Matthew Otten and Stephen K Gray. “Recovering noise-free quantum observables”. *Physical Review A* **99**, 012338 (2019).
 - [12] Tudor Giurgica-Tiron, Yousef Hindy, Ryan LaRose, Andrea Mari, and William J Zeng. “Digital zero noise extrapolation for quantum error mitigation”. *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)* Pages 306–316 (2020).
 - [13] Andre He, Benjamin Nachman, Wibe A. de Jong, and Christian W. Bauer. “Zero-noise extrapolation for quantum-gate error mitigation with identity insertions”. *Physical Review A* **102**, 012426 (2020).
 - [14] Zhenyu Cai. “Multi-exponential error extrapolation and combining error mitigation techniques for NISQ applications”. *npj Quantum Information* **7**, 1–12 (2021).
 - [15] Youngseok Kim, Christopher J. Wood, Theodore J. Yoder, Seth T. Merkel, Jay M. Gambetta, Kristan Temme, and Abhinav Kandala. “Scalable error mitigation for noisy quantum circuits produces competitive expectation values”. *Nature Physics* **19**, 752–759 (2023).
 - [16] Suguru Endo, Simon C Benjamin, and Ying Li. “Practical quantum error mitigation for near-future applications”. *Physical Review X* **8**, 031027 (2018).
 - [17] Bálint Koczor. “Exponential error suppression for near-term quantum devices”. *Physical Review X* **11**, 031057 (2021).
 - [18] William J Huggins, Sam McArdle, Thomas E O’Brien, Joonho Lee, Nicholas C Rubin, Sergio Boixo, K Birgitta Whaley, Ryan Babbush, and Jarrod R McClean. “Virtual distillation for quantum error mitigation”. *Physical Review X* **11**, 041036 (2021).
 - [19] Piotr Czarnik, Andrew Arrasmith, Lukasz Cincio, and Patrick J Coles. “Qubit-efficient exponential suppression of errors” (2021). url: <https://arxiv.org/abs/2102.06056>.
 - [20] Bálint Koczor. “The dominant eigenvector of a noisy quantum state”. *New Journal of Physics* **23**, 123047 (2021).
 - [21] Mingxia Huo and Ying Li. “Dual-state purification for practical quantum error mitigation”. *Physical Review A* **105**, 022427 (2022).
 - [22] Zhenyu Cai. “Resource-efficient purification-based quantum error mitigation” (2021). url: <https://arxiv.org/abs/2107.07279>.
 - [23] Alireza Seif, Ze-Pei Cui, Sisi Zhou, Senrui Chen, and Liang Jiang. “Shadow distillation: Quantum error mitigation with classical shadows for near-term quantum processors”. *PRX Quantum* **4**, 010303 (2023).
 - [24] Hong-Ye Hu, Ryan LaRose, Yi-Zhuang You, Eleanor Rieffel, and Zhihui Wang. “Logical shadow tomography: Efficient estimation of

- error-mitigated observables” (2022). url: <https://arxiv.org/abs/2203.07263>.
- [25] Thomas E. O’Brien, Stefano Polla, Nicholas C. Rubin, William J. Huggins, Sam McArdle, Sergio Boixo, Jarrod R. McClean, and Ryan Babbush. “Error mitigation via verified phase estimation”. *PRX Quantum* **2**, 020317 (2021).
- [26] Sam McArdle, Xiao Yuan, and Simon Benjamin. “Error-mitigated digital quantum simulation”. *Phys. Rev. Lett.* **122**, 180501 (2019).
- [27] Xavi Bonet-Monroig, Ramiro Sagastizabal, M Singh, and TE O’Brien. “Low-cost error mitigation by symmetry verification”. *Physical Review A* **98**, 062339 (2018).
- [28] Matthew Otten, Cristian L Cortes, and Stephen K Gray. “Noise-resilient quantum dynamics using symmetry-preserving ansatzes” (2019). url: <https://arxiv.org/abs/1910.06284>.
- [29] Zhenyu Cai. “Quantum error mitigation using symmetry expansion”. *Quantum* **5**, 548 (2021).
- [30] Yifeng Xiong, Soon Xin Ng, and Lajos Hanzo. “Quantum error mitigation relying on permutation filtering”. *IEEE Transactions on Communications* **70**, 1927–1942 (2022).
- [31] Nobuyuki Yoshioka, Hideaki Hakoshima, Yuichiro Matsuzaki, Yuuki Tokunaga, Yasunari Suzuki, and Suguru Endo. “Generalized quantum subspace expansion”. *Phys. Rev. Lett.* **129**, 020502 (2022).
- [32] Andrea Mari, Nathan Shammah, and William J Zeng. “Extending quantum probabilistic error cancellation by noise scaling”. *Physical Review A* **104**, 052607 (2021).
- [33] Lukasz Cincio, Yiğit Subaşı, Andrew T Sornborger, and Patrick J Coles. “Learning the quantum algorithm for state overlap”. *New Journal of Physics* **20**, 113022 (2018).
- [34] Lukasz Cincio, Kenneth Rudinger, Mohan Sarovar, and Patrick J. Coles. “Machine learning of noise-resilient quantum circuits”. *PRX Quantum* **2**, 010324 (2021).
- [35] Prakash Murali, Jonathan M. Baker, Ali Javadi-Abhari, Frederic T. Chong, and Margaret Martonosi. “Noise-adaptive compiler mappings for noisy intermediate-scale quantum computers”. In Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems. Page 1015–1029. ASPLOS ’19 New York, NY, USA (2019). Association for Computing Machinery.
- [36] Sumeet Khatri, Ryan LaRose, Alexander Poremba, Lukasz Cincio, Andrew T Sornborger, and Patrick J Coles. “Quantum-assisted quantum compiling”. *Quantum* **3**, 140 (2019).
- [37] Kunal Sharma, Sumeet Khatri, M. Cerezo, and Patrick J Coles. “Noise resilience of variational quantum compiling”. *New Journal of Physics* **22**, 043006 (2020).
- [38] Piotr Czarnik, Andrew Arrasmith, Patrick J. Coles, and Lukasz Cincio. “Error mitigation with Clifford quantum-circuit data”. *Quantum* **5**, 592 (2021).
- [39] Armands Strikis, Dayue Qin, Yanzhu Chen, Simon C Benjamin, and Ying Li. “Learning-based quantum error mitigation”. *PRX Quantum* **2**, 040330 (2021).
- [40] Ashley Montanaro and Stasja Stanisic. “Error mitigation by training with fermionic linear optics” (2021). url: <https://arxiv.org/abs/2102.02120>.
- [41] Joseph Vovrosh, Kiran E Khosla, Sean Greenaway, Christopher Self, Myungshik S Kim, and Johannes Knolle. “Simple mitigation of global depolarizing errors in quantum simulations”. *Physical Review E* **104**, 035309 (2021).
- [42] Miroslav Urbanek, Benjamin Nachman, Vincent R Pascuzzi, Andre He, Christian W Bauer, and Wibe A de Jong. “Mitigating depolarizing noise on quantum computers with noise-estimation circuits”. *Phys. Rev. Lett.* **127**, 270502 (2021).
- [43] Andrey Zhukov and Walter Pogosov. “Quantum error reduction with deep neural network applied at the post-processing stage”. *Quantum Information Processing* **21**, 93 (2022).
- [44] Haoran Liao, Derek S Wang, Iskandar Sitdikov, Ciro Salcedo, Alireza Seif, and Zlatko K Minev. “Machine learning for practical quantum error mitigation”. *Nature Machine Intelligence* **6**, 1478–1486 (2024).
- [45] Alejandro Sopena, Max Hunter Gordon, German Sierra, and Esperanza López. “Simulating quench dynamics on a digital quantum computer with data-driven error mitigation”. *Quantum Science and Technology* (2021).
- [46] Daniel Bultrini, Max Hunter Gordon, Piotr Czarnik, Andrew Arrasmith, M Cerezo, Patrick J Coles, and Lukasz Cincio. “Unifying and benchmarking state-of-the-art quantum error mitigation techniques”. *Quantum* **7**, 1034 (2023).
- [47] Angus Lowe, Max Hunter Gordon, Piotr Czarnik, Andrew Arrasmith, Patrick J. Coles, and Lukasz Cincio. “Unified approach to data-driven quantum error mitigation”. *Phys. Rev. Research* **3**, 033098 (2021).
- [48] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. “Variational quantum algorithms”. *Nature Reviews Physics* **3**, 625–644 (2021).
- [49] Samson Wang, Piotr Czarnik, Andrew Arrasmith, M. Cerezo, Lukasz Cincio, and Patrick J Coles. “Can error mitigation improve trainabil-

- ity of noisy variational quantum algorithms?”. *Quantum* **8**, 1287 (2024).
- [50] Shaojun Guo, Jinzhao Sun, Haoran Qian, Ming Gong, Yukun Zhang, Fusheng Chen, Yangsen Ye, Yulin Wu, Sirui Cao, Kun Liu, Chen Zha, Chong Ying, Qingling Zhu, He-Liang Huang, Youwei Zhao, Shaowei Li, Shiyu Wang, Jiale Yu, Daojin Fan, Dachao Wu, Hong Su, Hui Deng, Hao Rong, Yuan Li, Kaili Zhang, Tung-Hsun Chung, Futian Liang, Jin Lin, Yu Xu, Lihua Sun, Cheng Guo, Na Li, Yong-Heng Huo, Cheng-Zhi Peng, Chao-Yang Lu, Xiao Yuan, Xiaobo Zhu, and Jian-Wei Pan. “Experimental quantum computational chemistry with optimized unitary coupled cluster ansatz”. *Nature Physics* **20**, 1240–1246 (2024).
- [51] Hakop Pashayan, Oliver Reardon-Smith, Kamil Korzekwa, and Stephen D. Bartlett. “Fast estimation of outcome probabilities for quantum circuits”. *PRX Quantum* **3**, 020361 (2022).
- [52] David C. McKay, Christopher J. Wood, Sarah Sheldon, Jerry M. Chow, and Jay M. Gambetta. “Efficient z gates for quantum computing”. *Phys. Rev. A* **96**, 022330 (2017).
- [53] Minh C Tran, Kunal Sharma, and Kristan Temme. “Locality and error mitigation of quantum circuits” (2023). url: <https://arxiv.org/abs/2303.06496>.
- [54] Yu Zhang, Lukasz Cincio, Christian F. A. Negre, Piotr Czarnik, Patrick Coles, Petr M. Anisimov, Susan M. Mniszewski, Sergei Tretiak, and Pavel A. Dub. “Variational quantum eigensolver with reduced circuit complexity”. *npj Quantum Information* **8**, 96 (2022).
- [55] Ryan LaRose, Andrea Mari, Sarah Kaiser, Peter J. Karalekas, Andre A. Alves, Piotr Czarnik, Mohamed El Mandouh, Max H. Gordon, et al. “Mitiq: A software package for error mitigation on noisy quantum computers”. *Quantum* **6**, 774 (2022).
- [56] Cristina Cirstoiu, Silas Dilkes, Daniel Mills, Seyon Sivarajah, and Ross Duncan. “Volumetric Benchmarking of Error Mitigation with Qermit”. *Quantum* **7**, 1059 (2023).
- [57] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. “A quantum approximate optimization algorithm” (2014). url: <https://arxiv.org/abs/1411.4028>.
- [58] Dave Wecker, Matthew B. Hastings, and Matthias Troyer. “Progress towards practical quantum variational algorithms”. *Physical Review A* **92**, 042303 (2015).
- [59] Martín Larocca, Frédéric Sauvage, Faris M. Sbahi, Guillaume Verdon, Patrick J. Coles, and M. Cerezo. “Group-invariant quantum machine learning”. *PRX Quantum* **3**, 030341 (2022).
- [60] Johannes Jakob Meyer, Marian Mularski, Elies Gil-Fuster, Antonio Anna Mele, Francesco Arzani, Alissa Wilms, and Jens Eisert. “Exploiting symmetry in variational quantum machine learning”. *PRX Quantum* **4**, 010328 (2023).
- [61] Frederic Sauvage, Martin Larocca, Patrick J Coles, and M Cerezo. “Building spatial symmetries into parameterized quantum circuits for faster training”. *Quantum Science and Technology* **9**, 015029 (2024).
- [62] M. McKerns, L. Strand, T. Sullivan, A. Fang, and M.A.G. Aivazis. “Building a framework for predictive science”. *Proceedings of the 10th Python in Science Conference* (2011).
- [63] Michael McKerns, Patrick Hung, and Michael Aivazis. “mystic: highly-constrained non-convex optimization and UQ” (2009-).
- [64] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J Love, Alán Aspuru-Guzik, and Jeremy L O’Brien. “A variational eigenvalue solver on a photonic quantum processor”. *Nature Communications* **5**, 1–7 (2014).
- [65] Jarrod R. McClean, Kevin J. Sung, Ian D. Kivlichan, Yudong Cao, Chengyu Dai, E. Schuyler Fried, Craig Gidney, Brendan Gimby, Pranav Gokhale, Thomas Häner, Tarini Hardikar, Vojtěch Havlíček, Oscar Higgott, Cupjin Huang, Josh Izaac, Zhang Jiang, Xinle Liu, Sam McArdle, Matthew Neeley, Thomas O’Brien, Bryan O’Gorman, Isil Ozfidan, Maxwell D. Radin, Jhonathan Romero, Nicholas Rubin, Nicolas P. D. Sawaya, Kanav Setia, Sukin Sim, Damian S. Steiger, Mark Steudtner, Qiming Sun, Wei Sun, Daochen Wang, Fang Zhang, and Ryan Babush. “Openfermion: The electronic structure package for quantum computers” (2017). url: <https://arxiv.org/abs/1710.07629>.
- [66] Farrokh Vatan and Colin Williams. “Optimal quantum circuits for general two-qubit gates”. *Physical Review A* **69**, 032315 (2004).
- [67] Kathleen E Hamilton, Tyler Kharazi, Titus Morris, Alexander J McCaskey, Ryan S Bennink, and Raphael C Pooser. “Scalable quantum processor noise characterization”. In 2020 IEEE International Conference on Quantum Computing and Engineering (QCE). Pages 430–440. IEEE (2020).
- [68] Filip B Maciejewski, Zoltán Zimborás, and Michał Oszmaniec. “Mitigation of readout noise in near-term quantum devices by classical post-processing based on detector tomography”. *Quantum* **4**, 257 (2020).
- [69] Sergey Bravyi, Sarah Sheldon, Abhinav Kandala, David C McKay, and Jay M Gambetta. “Mitigating measurement errors in multiqubit experiments”. *Physical Review A* **103**, 042605 (2021).
- [70] Daniel Gottesman. “The heisenberg representation of quantum computers” (1998). url: <https://arxiv.org/abs/quant-ph/9807006>.

- [71] W. K. Hastings. “Monte Carlo sampling methods using markov chains and their applications”. *Biometrika* **57**, 97–109 (1970).
- [72] Lukasz Cincio, Kenneth Rudinger, Mohan Sarovar, and Patrick J. Coles. “Machine learning of noise-resilient quantum circuits”. *PRX Quantum* **2**, 010324 (2021).