

# Non-trivial symmetries in quantum landscapes and their resilience to quantum noise

Enrico Fontana<sup>1,2,3</sup>, M. Cerezo<sup>1,4</sup>, Andrew Arrasmith<sup>1</sup>, Ivan Rungger<sup>5</sup>, and Patrick J. Coles<sup>1</sup>

<sup>1</sup>Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA

<sup>2</sup>Department of Computer and Information Sciences, University of Strathclyde, 26 Richmond Street, Glasgow G1 1XH, UK

<sup>3</sup>National Physical Laboratory, Teddington TW11 0LW, UK

<sup>4</sup>Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, NM, USA

<sup>5</sup>National Physical Laboratory, Teddington, UK

Very little is known about the cost landscape for Parametrized Quantum Circuits (PQCs). Nevertheless, PQCs are employed in Quantum Neural Networks and Variational Quantum Algorithms, which may allow for near-term quantum advantage. Such applications require good optimizers to train PQCs. Recent works have focused on quantum-aware optimizers specifically tailored for PQCs. However, ignorance of the cost landscape could hinder progress towards such optimizers. In this work, we analytically prove two results for PQCs: (1) We find an exponentially large symmetry in PQCs, yielding an exponentially large degeneracy of the minima in the cost landscape. Alternatively, this can be cast as an exponential reduction in the volume of relevant hyperparameter space. (2) We study the resilience of the symmetries under noise, and show that while it is conserved under unital noise, non-unital channels can break these symmetries and lift the degeneracy of minima, leading to multiple new local minima. Based on these results, we introduce an optimization method called Symmetry-based Minima Hopping (SYMH), which exploits the underlying symmetries in PQCs. Our numerical simulations show that SYMH improves the overall optimizer performance in the presence of non-unital noise at a level comparable to current hardware. Overall, this work derives large-scale circuit symmetries from local gate transformations, and uses them to construct a noise-aware optimization method.

## 1 Introduction

The era of Noisy Intermediate Scale Quantum (NISQ)[1] computing has led to the emergence of novel algorithmic paradigms. Arguably, the leading role has been played by Parametrized Quantum Circuits (PQCs), which are exploited for both Variational Quantum Algorithms [2, 3, 4, 5, 6, 7,

8, 9, 10, 11, 12, 13, 14] and Quantum Neural Networks [15, 16, 17, 18, 19]. Training PQCs involves a hybrid quantum-classical optimization loop. Typically, the problem is encoded in a cost (or loss) function that is ideally efficient to evaluate on a quantum computer, but computationally expensive for a classical one. In practice, the cost function is estimated via measurements on a quantum computer which are usually post-processed on a classical device. While the quantum computer is used to evaluate the cost, a classical optimizer updates some (usually continuous) parameters associated with the quantum operations. PQCs with fixed gate structure are often referred to as a variational ansatz.

High performance classical optimizers are crucial to successfully train PQCs. To aid in optimizer selection and development, a fair amount of work has gone into determining the nature of quantum variational cost landscapes [20]. Important contributions include the development of analytical expressions for gradients of all orders [21, 22, 23, 24, 25], bounds on those derivatives [26], and even explicit functional forms for the expectation values computed with PQCs [27]. In addition, some light has recently been shed on the scaling of the gradient of quantum cost functions through a result known as barren plateaus [28, 29, 30, 24, 31, 32, 33]. This demonstrates that the landscape flattens exponentially with problem size for deep, unstructured PQCs, and also for shallow PQCs with global cost functions.

Even if one manages to avoid these barren plateaus, there is an additional difficulty of optimizing in the presence of the hardware noise that defines NISQ devices [34]. Hardware noise is expected to modify the cost landscape, and indeed it was recently shown to produce a novel kind of barren plateau whose impact increases with PQC depth [35]. Furthermore, while some models of hardware noise have been shown to leave the optimal parameters unchanged [36], this does not hold in general [37].

These results collectively reveal that the optimization of noisy PQCs presents many novel and unexpected challenges that must be addressed. This

has spawned the field of quantum-aware optimizers, where researchers are developing classical optimizers that are specifically tailored to the unusual landscape issues in the quantum setting. Examples include quantum natural gradient [38, 39], sequential function fitting [40], shot-frugal stochastic gradient descent [26, 41, 42], landscape modeling [43], and others [44]. Many of these optimizers are good at finding a local minimum, which has been shown to be accelerated by gradient information [45]. However, there remains the question of how to escape or move between such minima to find the global minimum.

In this work, we present a technique for training PQCs that we call Symmetry-based Minima Hopping (SYMh, pronounced ‘‘Sim’’). As the name suggests, SYMH is a method for hopping between local minima that exploits underlying symmetries in PQCs. The SYMH technique can be combined with other optimizers, such as those in Refs. [38, 39, 40, 26, 41, 42, 43, 44], to construct optimizers that search for minima that achieve lower costs by lowering the impact of noise. In this sense, our method is complementary to previous work, as classical local optimizers are easily integrated into the SYMH framework.

At the heart of our work is a novel understanding of symmetries and symmetry breaking in PQCs. In particular, we analytically prove that given some non-restrictive conditions on the ansatz, the cost landscape must have a certain periodicity, which gives rise to a large degeneracy of minima in the absence of noise. This is schematically shown in Fig. 1(a). In the noise-free setting, one can move between these degenerate minima with pulses that rotate the parameters by specific angles.

However, our second analytical result is that the symmetry can be broken by certain types of noise, specifically, non-unital noise (such as amplitude damping) and coherent noise. As a consequence, the degeneracy of various minima is lifted by noise, leading to landscapes like in Fig. 1(b). We denote this phenomenon noise-induced breaking of symmetries (NIBS).

The NIBS phenomenon, in turn, allows us to construct an optimizer where one exploits circuit symmetries to hop between local minima valleys. Because these valleys are no longer degenerate, such hopping potentially leads to lower cost values and mitigate the effect of noise by leading to better solutions. This is the idea behind our Symmetry-based Minima Hopping technique.

In what follows, we first lay out our general framework in the next section. We then discuss symmetries in PQCs in Section 3.1. In Section 3.2 we present a method to move between these symmetries that we call the  $\sigma$ -Pulse method. Section 4 considers the impact of noise and presents our results on the NIBS phenomenon. Section 5 presents our SYMH optimizer. Finally, Section 6 shows our numerical imple-

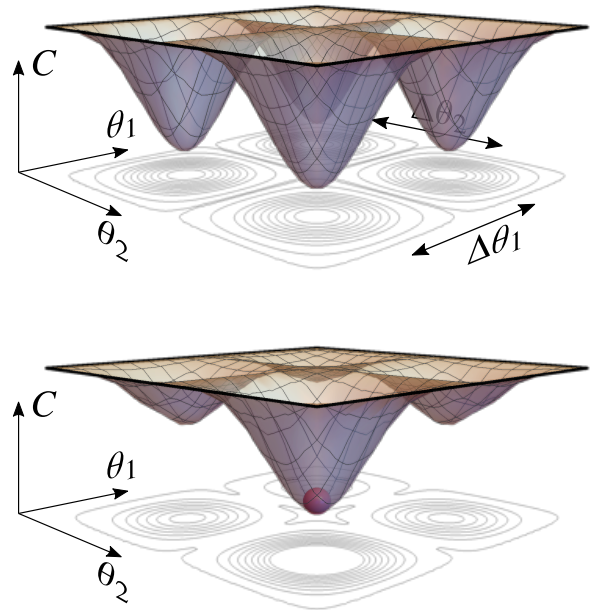


Figure 1: (a) Schematic diagram of a noise-free cost landscape. Here the cost function is periodic under some translation in the parameters of the ansatz  $V(\theta)$ , which leads to the global minima being degenerate. By randomly initializing and optimizing  $\theta$  one can converge to any of the degenerate minima. (b) Schematic diagram of the noisy cost landscape. The presence of quantum noise breaks the periodicity in the landscape, leading to some of the global minima that were degenerate in the noiseless scenario to become local minima. Now, when optimizing  $\theta$  one can get trapped in a local minimum.

mentations of SYMH, and demonstrates that SYMH leads to a significant improvement in the optimizer performance for various problems under realistic noise models.

## 2 General Framework

In this section we introduce the general framework for our work. Specifically, we consider here a generic quantum machine learning task where the goal is to minimize a parametrized cost function of the form

$$C(\theta) = \sum_{x=1}^S f_x(\rho_x, V(\theta)). \quad (1)$$

Here,  $\mathcal{S} = \{\rho_x\}_{x=1}^S$  is a training set of input states, and  $f_x$  is a function that determines the task at hand and which can be different for each input state. Moreover,  $V(\theta)$  is a PQC, and  $\theta$  are trainable parameters.

The cost function in (1) is in fact quite general and includes as special cases the cost functions for many important VQAs and QNNs. For example, in the Variational Quantum Eigensolver [3] we have  $S = 1$  and the cost  $C = \text{Tr}[HV(\theta)\rho V^\dagger(\theta)]$  is the expectation value of a given Hamiltonian  $H$ . Alternatively, in a binary classification problem, the cost

can be expressed as the mean-squared error  $C = \frac{1}{2K} \sum_x [y_x - \tilde{y}(\rho_x, V(\boldsymbol{\theta}))]^2$ , with  $y_x$  the true label, and  $\tilde{y}_x(\rho_x, V(\boldsymbol{\theta}))$  the predicted label for each state in the training set [16].

We here consider PQCs  $V(\boldsymbol{\theta})$  that can be expressed as the product of  $L$  unitaries as

$$V(\boldsymbol{\theta}) = V_L(\boldsymbol{\theta}_L) \cdots V_1(\boldsymbol{\theta}_1), \quad (2)$$

where  $\boldsymbol{\theta} = \{\boldsymbol{\theta}_l\}_{l=1}^L$  is a set of continuous parameters. Each unitary  $V_l(\boldsymbol{\theta}_l)$  can in turn be expanded as

$$V_l(\boldsymbol{\theta}_l) = \prod_{m_l=1}^{m_l} R_{\mu}(\theta_{m,l}) W_{m_l}, \quad (3)$$

where  $R_{\mu}(\theta) = e^{-i\sigma_{\mu}\theta}$  is a single qubit rotation, and where  $\sigma_{\mu} \in \{X_j, Y_j, Z_j\}_{j=1}^n$  is a Pauli operator on qubit  $j$ . Moreover,  $W_{m_l}$  denote unparametrized gates. For simplicity, we consider here that  $W_{m_l}$  are CNOTs or identities. That is,  $W_{m_l} \in \{\mathbb{1}_i, C_X^{ij}\}_{i,j \in \mathcal{C}}$ , where  $C_X^{ij}$  is a CNOT with control qubit  $i$  and target qubit  $j$ , and where  $\mathcal{C}$  is a graph of the qubit connectivity. Finally, take  $M$  to be the total number of controllable parameters in the PQC.

The PQC in (2)–(3) includes as special cases many ansatzes widely used in the literature. For instance, if  $V(\boldsymbol{\theta})$  is a hardware-efficient ansatz [46], then the set of available CNOTs is determined by the device connectivity. In addition,  $V(\boldsymbol{\theta})$  also includes as special cases the Quantum Alternating Operator Ansatz (QAOA) [5, 47], and the Unitary Coupled Cluster (UCC) [48, 49, 50] ansatz. Specifically, when implementing the QAOA or a UCC ansatz one usually performs first order Trotter approximations of unitaries of the form  $e^{-i\theta H}$ , where  $H$  is a Hermitian operator with an efficient decomposition in the Pauli basis. The latter then leads to a PQC that fits into the framework presently considered [35].

Given a PQC  $V(\boldsymbol{\theta})$ , we define its *buffered* version as follows.

**Definition 1** (Buffered PQC). *Let  $V(\boldsymbol{\theta})$  be a PQC as in (2) and (3). We define the buffered version of this PQC as a gate sequence*

$$V_B(\boldsymbol{\theta}, \boldsymbol{\gamma}) = U_B(\boldsymbol{\gamma})V(\boldsymbol{\theta}), \quad (4)$$

where  $U_B(\boldsymbol{\gamma})$  is the so-called buffer unitary given by

$$U_B(\boldsymbol{\gamma}) = \bigotimes_{j=1}^n e^{-iX_j\gamma_{2j}} e^{-iY_j\gamma_{2j-1}}. \quad (5)$$

As shown in Fig. 2, the buffer unitary is simply given by a tensor product of single qubit rotations around the  $x$  and  $y$  axes which are parametrized by the vector  $\boldsymbol{\gamma}$  of length  $2n$ .

We note that our main results, stated below, are valid for buffered PQCs. However, since  $V(\boldsymbol{\theta}) = V_B(\boldsymbol{\theta}, \mathbf{0})$  one can always trivially extend any PQC

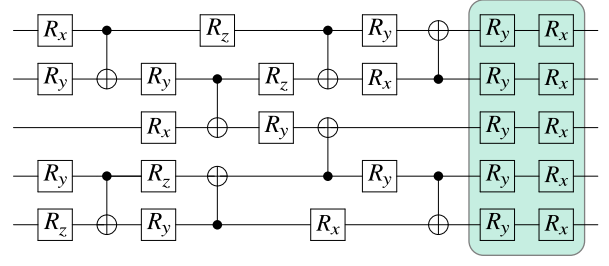


Figure 2: Schematic representation of a buffered PQC. The buffered version of any parametrized quantum circuit  $V(\boldsymbol{\theta})$  can be obtained by appending to the circuit a buffer unitary as described in (5). Here we have highlighted this buffer unitary.

to its buffered version. That is, any PQC of the form (2)–(3) can be considered as a buffered PQC with trivial rotation angles in the buffer unitary. In addition, we also remark that our results will also hold for any PQC where  $V(\boldsymbol{\theta})$  contains (at least) two single-qubit rotations about different axes in every qubit (not necessarily sequentially or in parallel). However, for the sake of simplicity in introducing the method, we consider the case where one appends a buffer unitary to the PQC.

Finally, we remark that in some practical settings the buffer unitary can be implemented virtually without any additional computational overhead. Whenever  $V(\boldsymbol{\theta})$  acts before measurement, the single-qubit rotations in the buffer layer can be absorbed into the measurement operator and executed classically by post-processing the measurement statistics. However, when the buffered unitary does not act prior to the measurements, such as when only a portion of the circuit is buffered, then  $U_B(\boldsymbol{\gamma})$  must be included.

### 3 $\sigma$ -Pulse method for finding parameter symmetries in PQCs

In this section, we first discuss symmetries in PQCs, which lead to degenerancies in the cost function landscape. We then present a method to move between different sets of parameters that are symmetric by propagating Pauli gates throughout the circuit, which we therefore call the  $\sigma$ -Pulse method.

#### 3.1 Symmetries in PQCs

As it was recently pointed out in [37], different sets of parameters  $\boldsymbol{\theta}$  in a PQC can lead to the same unitary being produced. In order to further analyze this phenomenon, we first introduce the following definition:

**Definition 2** (Parameter symmetries). *Let  $V(\boldsymbol{\theta})$  be a PQC. We say that two distinct sets of parameters  $\boldsymbol{\theta}$  and  $\tilde{\boldsymbol{\theta}}$  are symmetric if  $V(\boldsymbol{\theta})$  is equal to  $V(\tilde{\boldsymbol{\theta}})$  (up to a global phase).*

Let us here make two important remarks. First, note that Definition (2) implies that the structure of the circuit remains unchanged between  $V(\boldsymbol{\theta})$  and  $V(\tilde{\boldsymbol{\theta}})$ , as no gates in the circuit are being added or replaced; only their parameter values differ. Second, we remark that these parameter symmetries naturally translate into cost function landscape degenerancies. That is, given two symmetric sets of parameters  $\boldsymbol{\theta}$  and  $\tilde{\boldsymbol{\theta}}$  we have  $C(\boldsymbol{\theta}) = C(\tilde{\boldsymbol{\theta}})$ .

Note that there are many mechanics which can lead to symmetries in  $\boldsymbol{\theta}$ . For instance, they can arise from the wrapping symmetry in a rotation, i.e., from the fact that for any single qubit rotation we have  $R_\mu(\theta_\mu) = R_\mu(\theta_\mu + 2\pi)$ . Similarly, parameter symmetries can also be obtained from other types of mechanisms, such as commutation symmetries. Consider for example a two-qubit PQC composed of a CNOT preceded and followed by single qubit rotations about the  $z$  axis on the first qubit. That is,  $V(\theta_1, \theta_2) = (R_z(\theta_1) \otimes \mathbb{1}_2) \cdot C_X^{12} \cdot (R_z(\theta_2) \otimes \mathbb{1}_2)$ . Noting that  $[(R_z(\theta_1) \otimes \mathbb{1}_2), C_X^{12}] = 0$ , then it follows that  $V(\theta_1, \theta_2) = V(p\theta_1 + (1-p)\theta_2, (1-p)\theta_1 + q\theta_2)$  for any  $\theta_1$  and  $\theta_2$ , and for any  $p, q \in (0, 1)$ .

In the work we provide a general theory to analyze a class of non-trivial discrete parameter symmetries which were first reported in [37] and which we generalize here. Specifically, we introduce a method for finding and characterizing the following symmetries:

**Definition 3** ( $\sigma$ -Pulse symmetries). *Let  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  be a buffered PQC as in Definition 1.  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  possesses a  $\sigma$ -Pulse symmetry whenever, for any  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$ , there exists a set of  $\{\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\gamma}}\}$  related to  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  via*

$$\tilde{\theta}_j = (-1)^{p_j} \theta_j + q_j \pi, \quad \tilde{\gamma}_j = (-1)^{p'_j} \gamma_j + q'_j \pi, \quad (6)$$

for some  $p_j, q_j, p'_j, q'_j \in \{0, 1\}$ , such that  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  and  $\{\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\gamma}}\}$  are symmetric.

In the next section we present the so-called  $\sigma$ -Pulse method for determining  $\sigma$ -Pulse symmetries.

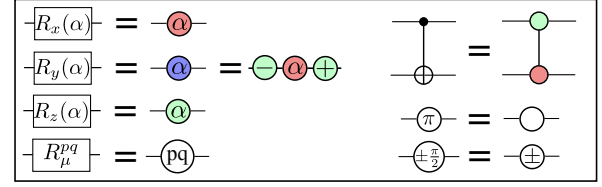
### 3.2 $\sigma$ -Pulse method

Here we introduce the  $\sigma$ -Pulse method which allow us to start from a set of parameters  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  of a buffered PQC and obtain a second set  $\{\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\gamma}}\}$  which are  $\sigma$ -Pulse symmetric to  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  according to Definition 3. This method is based on three basic steps: (1) The creation of the so-called  $\sigma$ -Pulses, (2) The propagation of said pulses through the circuit, and (3) The absorption of the  $\sigma$ -Pulses in the buffer unitary. The mathematical rules for these steps (described below in detail) are schematically shown in the ZX-calculus [51] notation of Fig. 3(a).

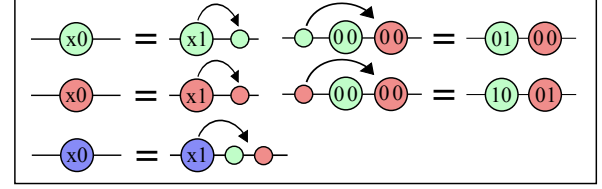
#### 3.2.1 Creation of $\sigma$ -Pulses

The first step of the  $\sigma$ -Pulse method is based on the fact that any single qubit rotation around a principal

#### a) ZX calculus notation



#### b) Creation and absorption rules



#### c) Commutation rules

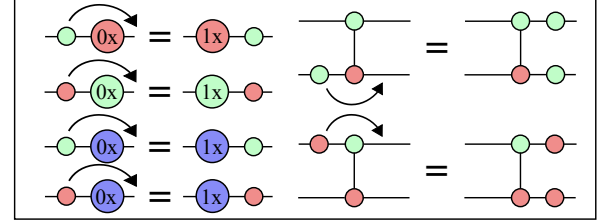


Figure 3:  $\sigma$ -Pulse method rules in the notation of ZX-calculus. (a) Notation for rotations around the principal axes, shifted rotations (of Eq. (8)), and for a CNOT gate. We additionally introduce notation for the  $\sigma$ -Pulses (rotation of angle  $\pi$ ), and for rotations of angles  $\pm \frac{\pi}{2}$ . (b) Rules for creating and absorbing  $\sigma$ -pulses. (c) Non-trivial commutation rules for  $\sigma$ -Pulses. We remark that in panels (b) and (c), the rules for a ( $i\sigma_y$ ) pulse can be derived from those of ( $i\sigma_x$ ) and ( $i\sigma_z$ ). In addition, in those panels an equal sign indicates that the unitaries are equal up to a global phase.

axis satisfies the following identity

$$\begin{aligned} R_\mu(\theta) &= e^{-i\theta\sigma_\mu/2} = e^{-i(\theta+\pi)\sigma_\mu/2} e^{i\pi\sigma_\mu/2} \\ &= R_\mu(\theta + \pi) (i\sigma_\mu) = R_\mu^{01} (i\sigma_\mu), \end{aligned} \quad (7)$$

where we defined the shifted rotations

$$R_\mu^{pq} = R_\mu((-1)^p\theta + q\pi), \quad (8)$$

for  $p, q \in \{0, 1\}$  and  $\mu \in \{x, y, z\}$ . As shown in Fig. 3(b), Eq. (7) implies that the angle of a single-qubit rotation in  $V(\boldsymbol{\theta})$  can be shifted by  $\pi$  at the expense of adding to the circuit a  $i\sigma_\mu$  gate, i.e., at the expense of creating a  $\sigma$ -Pulse. When Eq. (7) is employed to generate a pulse, we say that the gate is a *generator* of a *primary* pulse. Note that simply employing (7) changes the structure of the ansatz as we have added new gates to the circuit. For the structure of  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  to be preserved, the  $\sigma$ -Pulses need to be propagated through the circuit towards  $U_B(\boldsymbol{\gamma})$  where they be absorbed.

#### 3.2.2 Propagation of $\sigma$ -Pulses

Propagating the  $\sigma$ -Pulses through the circuit implies knowing how  $i\sigma_\mu$  commutes with all other gates in

the ansatz. The commutation of a  $\sigma_\mu$ -Pulse through a single qubit rotation  $R_\nu(\theta)$  is given by

$$R_\nu(\theta)(i\sigma_\mu) = i\sigma_\mu R_\nu((-1)^{\delta_{\mu\nu}}\theta) = i\sigma_\mu R_\nu^{\delta_{\mu\nu}0}. \quad (9)$$

Equation (9) shows that if  $\mu \neq \nu$ , the commutation of a pulse with a rotation can lead to said rotation picking up a minus sign. Moreover, the following identities provide the commutation rules between a pulse and a CNOT:

$$\begin{aligned} C_X(i\sigma_\mu \otimes \mathbb{1}) &= -i(i\sigma_\mu \otimes iX)C_X, \text{ for } \sigma_\mu = Y, X \\ C_X(\mathbb{1} \otimes i\sigma_\mu) &= -i(iZ \otimes i\sigma_\mu)C_X, \text{ for } \sigma_\mu = Y, Z \\ [C_X, (iZ \otimes \mathbb{1})] &= [C_X, (\mathbb{1} \otimes iX)] = 0, \end{aligned} \quad (10)$$

These commutation rules, which we illustrate in Fig. 3(c), show that commuting a  $\sigma$ -Pulse on the control (target) qubit through a CNOT can lead to the creation of a *secondary* pulse on the target (control) qubit, plus a global unobservable phase. For the gate structure of  $V_B(\theta, \gamma)$  to remain unchanged, the secondary pulses also need to be propagated towards the measurement, which in turn means that they can create additional secondary pulses.

### 3.2.3 Absorption of $\sigma$ -Pulses

Once all the primary and secondary pulses have been propagated to the buffer unitary, they can be absorbed by shifting the rotation in  $U_B(\gamma)$  angles as

$$\begin{aligned} R_x(\gamma')R_y(\gamma)(iX) &= R_x(\gamma' - \pi)R_y(-\gamma) = -R_x^{01}R_y^{10}, \\ R_x(\gamma')R_y(\gamma)(iY) &= R_x(\gamma')R_y(\gamma - \pi) = -R_x^{00}R_y^{01}, \end{aligned} \quad (11)$$

$$R_x(\gamma')R_y(\gamma)(iZ) = R_x(\gamma' - \pi)R_y(\pi - \gamma) = -R_x^{01}R_y^{11},$$

where we use the definition of the shifted rotations  $R_\mu^{pq}$  of (8). Here we remark that the minus signs on the right-hand side of (11) simply correspond to unobservable global phases. These absorption rules are shown in Fig. 3(b).

### 3.2.4 Parameter symmetries

Equations (9)–(11) provide the framework for determining symmetries in  $V_B(\theta, \gamma)$  with the  $\sigma$ -Pulse method. Given a set of angles  $\{\theta, \gamma\}$ , one can select any number of rotations to generate primary pulses. Once the primary and secondary pulses are propagated and absorbed in the buffer unitary, we define  $\{\tilde{\theta}, \tilde{\gamma}\}$  as the ensuing new set of angles. From Eqs. (7) and (11) it is straightforward to see that  $\{\tilde{\theta}, \tilde{\gamma}\}$  and  $\{\theta, \gamma\}$  are symmetric according to Eq (6) in Definition 3. In Fig. 4 we explicitly show this procedure.

Definition (3), and more specifically, Eq. (6), allows us to derive the following proposition, which is proved in the Appendix.

**Proposition 1** (Exponential Symmetry). *Let  $V_B(\theta, \gamma)$  be a buffered PQC. Then, for any  $\{\theta, \gamma\}$*

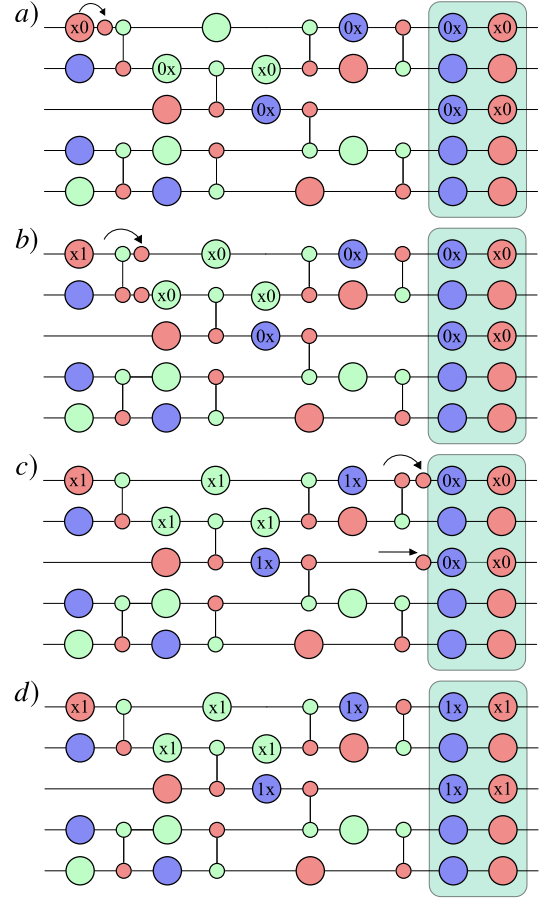


Figure 4: Schematic of the  $\sigma$ -Pulse method for finding parameter symmetries. We start with the buffered PQC of Fig. 2(a) with parameters  $\{\theta, \gamma\}$ . (a) A primary  $\sigma$ -Pulse is generated in the first  $R_x$  rotation acting on the first qubit. (b) While propagating the pulse through the CNOT gate, secondary pulses are created. (c) All the primary and secondary pulses are propagated to the buffer layer, where they can be absorbed. (d) Once the pulses are absorbed, we have a new set of parameters  $\{\tilde{\theta}, \tilde{\gamma}\}$  which are symmetric to  $\{\theta, \gamma\}$  according to the Definition 3.

there exists  $2^M$  sets of  $\sigma$ -Pulse symmetric parameters  $\{\tilde{\theta}, \tilde{\gamma}\}$  according to Definition (3). Each symmetric set can be characterized by a bitstring  $\beta$  of length  $M$  such that  $\beta_j = 0$  if  $\theta_j \in [0, \pi)$  and  $\beta_j = 1$  if  $\tilde{\theta}_j \in [\pi, 2\pi)$ . We denote as  $\mathcal{B}$  the set of such bitstrings.

Proposition 1 has several important implications. First, it shows that each point in the cost function landscape is exponentially periodic, as the cost is symmetric over the parameter translation  $\theta_j \rightarrow (\theta_j + \pi)$  for every  $j$ . This holds up to a sign change  $\theta_k \rightarrow -\theta_k$  in some other parameters with  $k \neq j$  and a correction rotation in the buffer layer parameters  $\gamma$ , due to the necessity to propagate and absorb the extra  $\sigma$ -Pulses. In particular, denoting as  $\{\theta_{\text{opt}}, \gamma_{\text{opt}}\}$  a set of parameters that minimize the cost  $C(\theta, \gamma)$  we have that the global minimum is  $2^M$ -fold degenerate.

In addition, Proposition 1 implies the next Corol-

lary.

**Corollary 1** (Effective Parameter Space Reduction). *Let  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  be a buffered PQC. Then, for every  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  there always exists a set of parameters  $\{\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\gamma}}\}$  which are  $\sigma$ -Pulse symmetric to  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$ , and which are such that  $\tilde{\theta}_j \in [0, \pi)$  for all  $j$ .*

The proof of Corollary (1) follows from Proposition 1 by taking  $\boldsymbol{\beta} = \mathbf{0}$ . This Corollary implies an exponential reduction of the effective hyperparameter space by restricting the domain of all angles in  $V(\boldsymbol{\theta})$  from  $[0, 2\pi)^M$  to  $[0, \pi)^M$ . Hence, all relevant features of the cost function landscape (including the global minima) can be found in  $[0, \pi)^M$ . We finally remark that the domain restriction in Corollary (1) is non-trivial as it does not arise from a wrapping symmetry in the rotation parameters (i.e., it does not arise from the fact that  $R_\mu(\theta) = R_\mu(\theta + 2\pi)$ ). Instead this domain reduction arises from the  $\sigma$ -Pulse symmetries.

## 4 Noise-induced lifting of the symmetries

In this section we analyze how noise affects the symmetries in  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  and hence the degeneracies in the cost landscape. Our main results are presented in the form of two theorems, with Theorem 1 analyzing the effect of unital Pauli noise, and Theorem 2 the effect of non-unital Pauli noise. We recall that unital Pauli noise channels include  $T2$  processes (i.e. dephasing channel), and depolarizing as special cases. On the other hand, non-unital Pauli noise channels include  $T1$  processes as a special case, i.e. the amplitude damping channel is a non-unital Pauli channel.

### 4.1 Unital Pauli noise

Consider the following definition:

**Definition 4** (Unital Pauli noise model). *We define the unital Pauli noise model as a process in which a unital Pauli channel acts after every layer of gates acting in parallel in  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$ .*

Here we recall that unital Pauli noise channels are completely positive trace-preserving maps  $\mathcal{P}_U$  whose superoperator is diagonal in the Pauli basis. The action of  $\mathcal{P}_U$  on an  $n$ -qubit Pauli operator  $X^{\mathbf{a}}Z^{\mathbf{b}}$  is given by

$$\mathcal{P}_U(X^{\mathbf{a}}Z^{\mathbf{b}}) = p_{\mathbf{ab}}X^{\mathbf{a}}Z^{\mathbf{b}}, \quad (12)$$

where  $p_{\mathbf{00}} = 1$ , and where we assume that  $-1 \leq p_{\mathbf{ab}} \leq 1 \forall \mathbf{a}, \mathbf{b}$ . Here  $\mathbf{a}, \mathbf{b} \in \{0, 1\}^{\otimes n}$  are bitstrings of length  $n$ , and where we employ the notation

$$X^{\mathbf{a}} = X_1^{a_1} \otimes \cdots \otimes X_n^{a_n}, \quad Z^{\mathbf{b}} = Z_1^{b_1} \otimes \cdots \otimes Z_n^{b_n}. \quad (13)$$

As explicitly shown in the Appendix, the following theorem holds.

**Theorem 1** (Symmetry preservation). *Let  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  be a buffered PQC as in Definition 1. Then, the  $\sigma$ -Pulse parameter symmetries in  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  are preserved under the action of the unital Pauli noise model in Definition 4.*

Theorem 1 shows that the parameter symmetries in  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  arising from  $\sigma$ -Pulse symmetries are completely preserved by the action of unital Pauli noise channels which includes as special cases the action of local (or global) depolarizing channels, as well as dephasing channels.

In addition, Theorem 1 implies that the degeneracy in the cost function landscape also remains unchanged. Particularly, we then know that the optimal parameters  $\{\hat{\boldsymbol{\theta}}_{\text{opt}}, \hat{\boldsymbol{\gamma}}_{\text{opt}}\}$  leading to the global minimum of the noisy cost function will still be  $2^M$ -fold degenerate in  $[0, 2\pi)$ . That is, starting from  $\{\hat{\boldsymbol{\theta}}_{\text{opt}}, \hat{\boldsymbol{\gamma}}_{\text{opt}}\}$ , all the symmetric parameters obtained from any bitstring in  $\mathcal{B}$  will have the same energy. In a practical scenario Theorem 1 implies that the minimum cost achievable from randomly initializing the parameters  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  will be independent of the bitstring  $\boldsymbol{\beta}$  characterizing the initial point.

Note however that for general cost functions the presence of quantum noise can change the cost landscape such that the optimal parameters of the noisy cost function are different than the ones for the noiseless case [4]. That is, one generally has  $\{\hat{\boldsymbol{\theta}}_{\text{opt}}, \hat{\boldsymbol{\gamma}}_{\text{opt}}\} \neq \{\boldsymbol{\theta}_{\text{opt}}, \boldsymbol{\gamma}_{\text{opt}}\}$ . For the special cases when  $\{\hat{\boldsymbol{\theta}}_{\text{opt}}, \hat{\boldsymbol{\gamma}}_{\text{opt}}\} = \{\boldsymbol{\theta}_{\text{opt}}, \boldsymbol{\gamma}_{\text{opt}}\}$  we say that the cost has *optimal parameter resilience*. This phenomenon has been analyzed in [36] for the problem of variational quantum compiling.

In addition, we also know that the value of the noisy cost function evaluated at the optimal parameters  $\hat{C}(\hat{\boldsymbol{\theta}}_{\text{opt}}, \hat{\boldsymbol{\gamma}}_{\text{opt}})$  can also change due to the presence of noise. Here  $\hat{C}$  denotes the noisy cost function. In fact, let us consider a cost function of the form  $C(\boldsymbol{\theta}, \boldsymbol{\gamma}) = \text{Tr}[OV(\boldsymbol{\theta})\rho V^\dagger(\boldsymbol{\theta})]$ , where  $O$  is a Hermitian operator, and assume that the parameters  $\{\boldsymbol{\theta}_{\text{opt}}, \boldsymbol{\gamma}_{\text{opt}}\}$  yield the operator's ground state. Then we have trivially that

$$C(\boldsymbol{\theta}_{\text{opt}}, \boldsymbol{\gamma}_{\text{opt}}) \leq \hat{C}(\hat{\boldsymbol{\theta}}_{\text{opt}}, \hat{\boldsymbol{\gamma}}_{\text{opt}}). \quad (14)$$

In general, the output state of the PQC converges to the fixed point of the noise model [52], and so the cost function will be increasingly different as noise increases.

### 4.2 Non-unital Pauli noise

Let us here analyze the effect of non-unital Pauli noise on the  $\sigma$ -Pulse symmetries. Hence, consider the following definition

**Definition 5** (Non-unital Pauli noise model). *We define the non-unital Pauli noise model as a process*

in which a non-unital Pauli channel acts after every layer of gates acting in parallel in  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$ .

Here we recall that non-unital Pauli noise channels are completely positive trace-preserving maps  $\mathcal{P}_{\text{NU}}$  whose action on the identity operator is

$$\mathcal{P}_{\text{NU}}(\mathbb{1}) = \mathbb{1} + \sum_{(a,b) \neq (0,0)} d_{ab} X^a Z^b. \quad (15)$$

On the other hand, its action on all other Pauli operators is given by

$$\mathcal{P}_{\text{NU}}(X^a Z^b) = q_{ab} X^a Z^b. \quad (16)$$

As explicitly shown in the Appendix, the following theorem holds.

**Theorem 2** (Symmetry breaking). *Let  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  be a buffered PQC as in Definition 1. Then, the  $\sigma$ -Pulse parameter symmetries in  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  can be broken under the action of a non-unital Pauli noise model.*

From Theorem 2 we have that given two sets of symmetric parameters  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  and  $\{\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\gamma}}\}$ , then there exists some non-unital noise such that we have  $V(\boldsymbol{\theta}, \boldsymbol{\gamma}) \neq V(\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\gamma}})$ . This implies that the  $2^M$ -fold symmetry of the optimal noisy parameters, and hence the degeneracy in the cost landscape, is broken.

In particular we now have that some of the (previously) global minima are transformed into local minima. In practical terms, due to this degeneracy breaking in the cost landscape not all randomly initialized  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  will converge to the global minima. In the next section we show how this effect can be mitigated by exploiting the knowledge of parameter symmetries to construct an optimizer.

We remark that the proof for Theorem 2 in the Appendix is valid for more general noisy channels that include as special case non-unital Pauli noise channels. For instance, we show that coherent error, such as qubit drift, can also break the  $\sigma$ -Pulse symmetries.

## 5 Symmetry-based Minima Hopping (SYMH) optimizer

Here we present the Symmetry-based Minima Hopping (SYMH) optimizer, which is meant to be employed in the presence of non-unital quantum noise. As its name indicates SYMH employs the  $\sigma$ -Pulse symmetries to hop around the degeneracy-broken landscape and attempt to find the minima that are less sensitive to noise. As further explained below, the strength of SYMH is that it should be considered a general tool which can be implemented along with other optimization and error mitigation techniques.

Consider the problem of minimizing the cost  $C(\boldsymbol{\theta}, \boldsymbol{\gamma})$  function in Eq. (1). As shown in Fig. 5(a), a common strategy is to randomly initialize the parameters  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  and employ a classical optimizer which

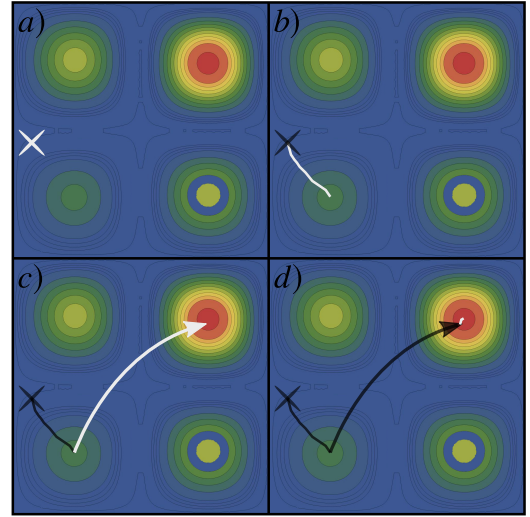


Figure 5: Main idea behind the SYMH optimizer. The contour plots correspond to the cost landscape of Fig. 1(b), where the presence of non-unital quantum noise has broken the landscape degeneracy. a) The parameters are random initialized. b) By employing a classical optimizer we can determine a cost minimizing direction. In this case, the minima to which the optimizer converges is a local minima which used to correspond to a global minima in the noiseless case. c) Using the  $\sigma$ -Pulse method we can hop in the landscape and land in the vicinity of another minima. d) By performing an optional second optimization we can find the global minima of the problem.

takes as input the value of the cost (or its gradients) to determine a cost minimizing direction. Usually one optimizes until some stopping criteria has been met, at which point one hopes that the minima reached corresponds to a global optima. We call the parameters obtained at the end of the optimization as  $\{\boldsymbol{\theta}_f, \boldsymbol{\gamma}_f\}$ , where the  $f$  stands for final.

However, as shown in Fig. 5(b), randomly initializing the parameters can lead to the optimizer getting trapped in local minimum which used to correspond to global minimum in the noiseless scenario. In this case, one can attempt to find the global minimum by taking final parameters  $\{\boldsymbol{\theta}_f, \boldsymbol{\gamma}_f\}$  and employing the  $\sigma$ -Pulse method (i.e. using Eq. (6)) to obtain a new set of parameters  $\{\tilde{\boldsymbol{\theta}}_f, \tilde{\boldsymbol{\gamma}}_f\}$ . As depicted in Fig. 5(c) this will effectively lead to a hop in the cost-landscape whose ending point can be in the well of another minima.

Since the cost landscape degeneracy is broken, then  $\{\tilde{\boldsymbol{\theta}}_f, \tilde{\boldsymbol{\gamma}}_f\}$  might not correspond to a critical point: one therefore has to re-optimize. This is schematically shown in Fig. 5(d), where the hop leads to the vicinity of the minimum and additional optional optimization could be needed. Note that in general this optional optimization will only require a small number of circuit evaluations, and hence will not add significant overhead to the optimization.

Here we remark that there are several possibilities

for where the SYMH takes us in the landscape. As previously mentioned, in the best case scenario the hopping can lead to the vicinity of a minimum whose cost function value is smaller than  $C(\boldsymbol{\theta}_f, \boldsymbol{\gamma}_f)$ . Here SYMH was successful as it allowed to mitigate the effect of noise and improve the quality of the solutions. It might also happen that one lands in the vicinity of a minimum whose cost function value is larger or equal than  $C(\boldsymbol{\theta}_f, \boldsymbol{\gamma}_f)$ . In this case one simply rejects this particular hop and use Eqs. (6) to hop to a new set of  $\{\tilde{\boldsymbol{\theta}}_f, \tilde{\boldsymbol{\gamma}}_f\}$  with different  $\beta \in \mathcal{B}$ . Finally, if the cost landscape has shifted in such a way that hop does not lead to the vicinity of a minima, one can still perform the second optimization. This would simply correspond to an optimization starting from a new random seed. In all cases, the overhead added by employing SYMH does not change the overall complexity of the optimization.

Note that this high-level description of SYMH is intended as a template than can be employed in many different scenarios. Due to the versatility of the method we do not intend to present here an all-descriptive way of employing SYMH but rather to introduce it as a general method which can be coupled to other optimization and error mitigation techniques. In what follows we present different SYMH-based optimization techniques.

## 5.1 Parameter sweeping method

One of the main challenges that can arise when employing SYMH is the exponentially large number of possible hops one can take (arising from the  $2^M$  bit-strings in  $\mathcal{B}$ ). In this section we present a technique called the *sweeping method* in which instead of exploring all hops, one instead simply explores a reduced sub-set of hops which adds an overhead at most in  $\mathcal{O}(M)$ .

In the sweeping method one starts with  $\{\boldsymbol{\theta}_f, \boldsymbol{\gamma}_f\}$  and employs SYMH to obtain a new set  $\{\tilde{\boldsymbol{\theta}}_f, \tilde{\boldsymbol{\gamma}}_f\}$  such that  $\boldsymbol{\theta}_f$  and  $\tilde{\boldsymbol{\theta}}_f$  differ only in the first parameter. That is, the first parameter in the sets  $\boldsymbol{\theta}_f$  and  $\tilde{\boldsymbol{\theta}}_f$  are related according to (6). This guarantees that all the parametrized gates in  $V(\boldsymbol{\theta})$  except for the first one remain the same. Then, as previously described one performs a second optimization to find the cost function value at  $\{\tilde{\boldsymbol{\theta}}_f, \tilde{\boldsymbol{\gamma}}_f\}$ , which determines if the shift is accepted or rejected. This procedure is sequentially repeated by sweeping  $n_s$  times through all  $M$  parameters. We refer the reader to Algorithm 1 for a more detailed description. One of the main advantages of this method is that it allows us to identify the parameters that, when shifted, yield the biggest improvement in terms of cost function minimization.

---

### Algorithm 1: $\sigma$ -PULSE-BASED

PARAMETER SWEEPING OPTIMIZATION  
METHOD

---

**Input:** Buffered PQC with parameters  $\boldsymbol{\alpha} = (\boldsymbol{\theta}, \boldsymbol{\gamma})$  and cost function  $C(\boldsymbol{\alpha})$ ; local optimizer `opt` returning the parameters of a local minimum given a starting point; parameter symmetry `symh` with input a list of indices of parameters to be shifted; number of sweeps  $n_s$ .

**Output:** Converged  $C_f$  and corresponding parameters  $\boldsymbol{\alpha}_f$ .

**Init:** Choose  $\boldsymbol{\alpha}_0$  at random;  $\boldsymbol{\alpha}_f \leftarrow \mathbf{0}$ ;  $C_f \leftarrow 0$ ; `pulse_lst`  $\leftarrow \{\}$ .

- 1 Optimize with `opt` with initial value  $\boldsymbol{\alpha}_0$  and store result in  $\boldsymbol{\alpha}_f$ ;  $C_f \leftarrow C(\boldsymbol{\alpha}_f)$ .
  - 2 **repeat**
  - 3     **for** each parameter in circuit not in `pulse_lst` or in buffer **do**
  - 4         Use `symh` with input  $\boldsymbol{\alpha}_f$  to shift the corresponding parameter obtaining  $\tilde{\boldsymbol{\alpha}}$ ;
  - 5         (Optional) Use `opt` with input  $\tilde{\boldsymbol{\alpha}}$  to find  $\tilde{\boldsymbol{\alpha}}_f$ ;
  - 6         Evaluate  $C(\tilde{\boldsymbol{\alpha}}_f)$ .
  - 7     At end of cycle append the best parameter index to `pulse_lst`;  $\boldsymbol{\alpha}_f \leftarrow \tilde{\boldsymbol{\alpha}}_f$ ;
  - 8      $C_f \leftarrow C(\tilde{\boldsymbol{\alpha}}_f)$ .
  - 8 **until** no more improvement in  $C_f$ ; at most  $n_s$  times.
  - 9 (Optional) Do final round of optimization with `opt`.
  - 10 **return**  $C_f, \boldsymbol{\alpha}_f$ .
- 

## 5.2 Using SYMH for ansatz symmetry breaking and landscape exploration

Let us now discuss how to implement SYMH to improve the solution quality in problems where the ansatz encodes some additional symmetry beyond that of the  $\sigma$ -Pulses. In general those symmetries are translate into constraints in the parameters of  $V(\boldsymbol{\theta})$ . Specifically, we analyze the possibility of using SYMH to break those additional constraints and improve the solution quality via the parameter hops obtained through the  $\sigma$ -Pulse method.

Unstructured ansatzes such as the hardware efficient ansatz [46] have been widely implemented in the literature for problems in which one has little to no information about the solution of the problem. However, there are many tasks in which one possesses knowledge which can be employ to construct the so-called physically-inspired ansatzes. Such information can come in the form of a specific symmetry that the ansatz must preserve [53], an adiabatic transformation that must be followed [5, 47] (as in QAOA), or the operators that the ansatz must contain [48, 49, 50]



(as in UCC). Preserving these additional symmetries during the parameter training can guarantee that the ansatz explores a sub-space of states related to the solution of the problem.

In most cases, these additional symmetries are translated into parameter constraints which the circuit description of  $V(\boldsymbol{\theta})$  must obey. For instance, in QAOA all the parameters in a given mixing of driving layer are correlated [54]. Note that in the SYMH formalism this corresponds to exploring certain subspaces of  $\mathcal{B}$  where the bistrings  $\boldsymbol{\beta}$  respect the ansatz symmetry.

In the presence of noise, however, it could happen that one can obtain a higher quality solution by starting from the set of ansatz constraint-preserving parameters  $\{\boldsymbol{\theta}_f, \boldsymbol{\gamma}_f\}$ , and using SYMH to hop to an ansatz constraint-breaking set of parameters. In this case, SYMH allows us to explore circuit configurations inaccessible to the original circuit structure.

### 5.3 The Quantum Alternating Operator Ansatz

We explicitly characterize the symmetries possessed by a Quantum Alternating Operator Ansatz (QAOA) [47], without a buffer layer. To recap, the QAOA, which generalises the ansatz in the Quantum Approximate Optimization Algorithm [5], has a layered structure, each layer being composed of two unitaries: a problem unitary  $U_P(\beta_i) = e^{-i\beta_i H_P}$ , where  $H_P$  is the problem Hamiltonian (consisting of  $Z$ -terms only) and a mixing unitary  $U_M(\gamma_i) = e^{-i\gamma_i H_M}$ , where typically  $H_M = \sum_j X_j$ . The parameters of QAOA can therefore be represented as two vectors  $\{\boldsymbol{\beta}, \boldsymbol{\gamma}\}$  each corresponding to each type of unitary. Now let us consider the effect of shifting one of these parameters by  $\pi$ .

If the parameter belongs to a mixing unitary, the shift will be cancelled by the creation of a  $\sigma$ -Pulse in the  $X$  direction on all the qubits. It can be shown that the commutation of any number of  $X$  pulse through a gate consisting of  $Z$  terms of any order generates no additional pulses, but leads to the parameter of the gate acquiring a negative sign. Therefore, the  $X$  pulses generated by such a shift can only be annihilated by a restoring shift of another mixing parameter. Therefore, a symmetry is the following:

$$\begin{aligned}\gamma_i &\rightarrow \gamma_i + \pi \\ \gamma_j &\rightarrow \gamma_j + \pi, \quad i < j \\ \beta_k &\rightarrow -\beta_k, \quad \forall i < k \leq j\end{aligned}$$

If instead we focus on the problem unitary, we see that the parameter shift will create, on a given qubit, one  $Z$  pulse for each  $Z$  acting on that qubit. Since the mixing unitary presents one  $X$  rotation for each qubit, one concludes that, in order to be able to commute all of the pulses past a mixing unitary with only a

sign modification of its parameter, there must be the same number of  $Z$  pulses on all the qubits. Therefore, the ansatz will only possess such a symmetry if the problem Hamiltonian features the same number of  $Z$ s on each qubit. This happens, for example, in MaxCut problems on  $n$ -regular graphs. In this specific case, when  $n$  is odd each shift will generate one total  $Z$  per qubit, and for any  $i, j$  with  $i < j$  the symmetries are:

$$\begin{aligned}\beta_i &\rightarrow \beta_i + \pi, \quad \beta_j \rightarrow \beta_j + \pi, \\ \gamma_k &\rightarrow -\gamma_k \quad \forall k \quad i < k \leq j.\end{aligned}$$

Otherwise, for  $n$  even all the pulses will cancel and the symmetries will simply be:

$$\beta_i \rightarrow \beta_i + \pi \quad \forall i.$$

Identifying these symmetries is significant, because they imply, if not a complete parameter space reduction, that restricting the range of some parameters will not affect the result of the algorithm.

## 6 Implementations

In this section we present heuristic results where we employ SYHM to improve the solution quality of VQAs in the presence of quantum noise. Specifically, we simulate VQAs for two implementations: variational quantum compiling, and for a Variational Quantum Eigensolver problem.

### 6.1 Variational Quantum Compiling

Quantum compiling [55, 56, 57] refers to the task of transforming a high-level algorithm into a low-level code that a quantum hardware can efficiently implement. In the near-term, one of the main applications for quantum compiling is to transforming a unitary with a deep quantum circuit description into a shorter depth gate sequence which mitigates the effect of noise. Several Variational Quantum Compiling (VQC) [58, 59, 7] architectures have been recently introduced where one trains the parameters in a short-depth PQC so that its outputs approximate those of a target unitary  $U$ .

Here we consider a quantum compiling application where the goal is to train the parameters in the PQC so that  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})|\mathbf{0}\rangle = U|\mathbf{0}\rangle$ , where  $U$  is the  $W$ -state preparation circuit for three qubits (see [36] for an explicit circuit), and where  $|\mathbf{0}\rangle = |0\rangle^{\otimes 3}$  is the all-zero state. Explicitly,  $U|\mathbf{0}\rangle = |W\rangle$ , where  $|W\rangle$  is the three qubit  $W$ -state. For simplicity, we consider the cost function

$$C = 1 - \text{Tr} \left[ U|\mathbf{0}\rangle\langle\mathbf{0}|U^\dagger V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})|\mathbf{0}\rangle\langle\mathbf{0}|V_B^\dagger(\boldsymbol{\theta}, \boldsymbol{\gamma}) \right], \quad (17)$$

which vanishes if  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})|\mathbf{0}\rangle = |W\rangle$  (up to a global phase).

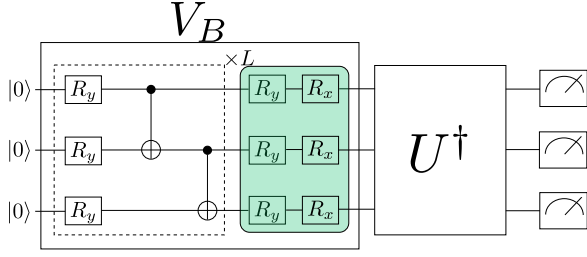


Figure 6: Circuit employed for the  $W$ -state quantum compilation implementation. Here we have indicated with a box the ansatz for  $V_B(\theta, \gamma)$ . By measuring the probability of all qubits being on zero  $P(0)$ , the cost can be computed as  $C = 1 - P(0)$ . Here the buffer layer is only composed of rotations about the  $y$  axis, as those are sufficient.

As outlined in Fig. 6, for the buffered PQC, we choose a layered hardware-efficient ansatz of the form

$$U(\theta, \gamma) = U_B(\gamma) \left( \prod_{i=1}^L U_i(\theta_i) \right), \quad (18)$$

where each  $U_i(\theta_i)$  consists of single qubit rotations followed by CNOTs. Note that here  $U_B(\gamma)$  is only composed of  $R_y$  rotations as we can only create  $\sigma_y$  pulses.

In Fig. 7 we present our numerical results. Here we employed a noisy quantum circuit simulator with realistic amplitude damping noise acting after every gate (including idle gates) obtained from the average T1 and T2 parameters and gate times of the `ibmq_melbourne` quantum computer. For gradient descent we used the COBYLA [60] optimizer, initialized at random angles. In addition, we simulated the circuit with an ansatz composed of  $L = 1, 2, 3$  layers. In a noiseless scenario, a single layer is not enough to prepare the  $W$  state, while  $L = 2, 3$  can reproduce the desired target state. For each number of layers we ran 100 randomly initialized simulations, and once the first optimization was completed we implemented the parameter sweeping method of Algorithm 1 with  $n_s = 4$  maximum sweeps.

As seen in Fig. 7, for all values of  $L$  employing SYMH leads to a systematic improvement in the cost function value. This improvement can be measured by taking either the best run (out of 100) or by taking the average of all runs before and after SYMH. In addition we can see that the improvement increases with  $L$  as circuits with a longer depths accumulate more noise, hence leading to a larger possible improvement.

## 6.2 Variational Quantum Eigensolver

In this section we present our results for a Variational Quantum Eigensolver (VQE) implementation. Here, the goal is to train a PQC such that  $V_B(\theta, \gamma)$  prepares the ground-state of a given Hamiltonian  $H$ . Specifically, we consider  $H$  to be the  $SU(2)$ -symmetric

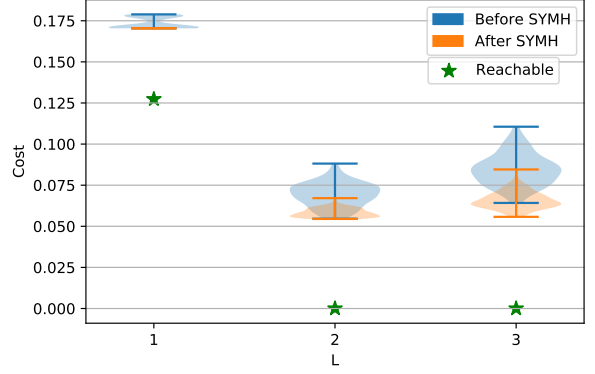


Figure 7: Numerical results for the  $W$ -state quantum compiling implementation. We show cost function value versus the number of layers  $L$  in  $V(\theta)$  before and after implementing SYMH, for a circuit subject to a realistic noise model with amplitude damping noise. The stars indicate the cost function value reachable in the absence of noise. For each  $L$ , we show the spread of the cost function value results for 100 randomly initialized instances of the optimization, with the horizontal lines indicating the best and worst run.

Heisenberg  $XXX$  model on  $n$  qubits

$$H_{XXX} = \sum_{i=1}^n \sigma_x^i \sigma_x^{i+1} + \sigma_y^i \sigma_y^{i+1} + \sigma_z^i \sigma_z^{i+1}. \quad (19)$$

Where we assume periodic boundary conditions so that  $n+1 \equiv 1$ . Here the cost function is simply given by

$$C = \text{Tr}[HV(\theta)|\psi\rangle\langle\psi|V^\dagger(\theta)], \quad (20)$$

where  $|\psi\rangle$  is an efficiently preparable input state.

For the PQC ansatz we consider a subclass of the QAOA called Hamiltonian Variational Ansatz (HVA), first introduced in [61]. Specifically, we follow the circuit structure for HVAs of [62], where we split  $H_{XXX}$  into two summations with the  $i$  index being even and odd. That is,  $H_{XXX} = H_{\text{odd}} + H_{\text{even}}$  so that the ansatz can be expressed as

$$V(\theta) = \prod_{i=1}^L e^{-i\theta_i H_{\text{even}}/2} e^{-i\theta_i H_{\text{odd}}/2}. \quad (21)$$

In addition, here we choose the initial state to be a tensor product of the Bell state  $|\psi\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)^{\otimes n/2}$ , which is the ground state of  $H_{\text{even}}$ . As shown in Fig. 8, the circuit description of  $V(\theta)$  can be obtained from a first order Trotter expansion of the  $e^{-i\theta_i H_{\text{even}}/2}$  and  $e^{-i\theta_i H_{\text{odd}}/2}$  unitaries and hence features alternating layers of  $XX$ ,  $YY$  and  $ZZ$  interactions, first on odd qubits and then on even qubits.

Let us remark that the parameters in  $V(\theta)$  encode additional problem symmetries as all the gates in a layer are constrained to being identical. To show that SYMH can be used to break these constraints and

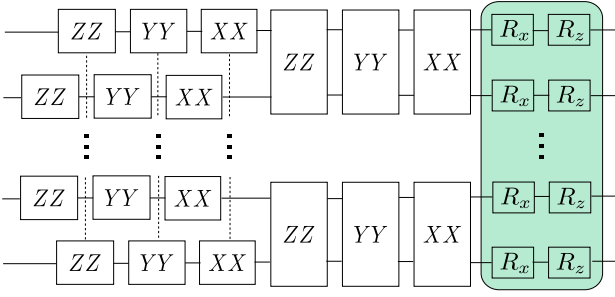


Figure 8: Single layer of the Hamiltonian Variational Ansatz for the  $XXX$  model. Here, the gates  $XX$  indicate a unitary of the form  $e^{-i\theta X_i X_{i+1}}$ . We employ a similar notation for  $YY$  and  $ZZ$ . The angles in the buffer layer are set to zero during the initial optimization to preserve the ansatz constraints.

improve the solution quality in the presence of noise, our heuristics where perform with the optimization schedules of Fig. 9. First, we set the parameters in the buffer layer to be  $\gamma = \mathbf{0}$  so that  $U_B(\theta) = \mathbb{1}$ . Then, we randomly initialize the  $\theta$  and optimize them respecting their correlation (see Fig. 9(1)). As indicated in Fig. 9(2), we then implement SYMH to break the VHA constraints, and hop in the landscape. This hopping can be followed by a second optimization, which we call a “free” optimization, where the VHA constraints are broken and all parameters are independently optimized (see Fig. 9(3)). To verify that the improvements arise from the SYMH and not from the free optimization, for each run we additionally perform a free optimization without SYMH (Fig. 9(4)).

The results from our numerical simulations are presented in Fig. 10. Here we considered VQE problems with  $n = 4, 6, 8, 10$  qubits and where the ansatz is composed of a single layer. Moreover, we employ the same noise model and optimizer as the one described in the previous section. For each  $n$  we ran 100 instances of the optimization. In Fig. 10(a) we present the best run for each number of qubits and for the different optimization method in Fig. 9. As shown, one can always improve the solution quality by breaking the ansatz constrains. However, the best solution is always achieved when employing the SYMH. Meaning that the optimal improvement follows from hoping in the landscape.

A more detailed comparison of the cost function improvement for the different methods is presented in Fig. 10(b), where the improvement is defined as

$$\text{improvement \%} = 100 \times \frac{E_f - E_{HVA}}{E_{GS}}. \quad (22)$$

Here,  $E_{GS}$  denotes the true ground state energy  $E_{HVA}$  is the energy obtained from (1) in Fig. 9 and  $E_f$  is the final energy from the optimization schemes (2), (3) and (4) of Fig. 9. As shown in Fig. 10(b), employing SYMH and breaking the parameter constrains always seems to leads to the best improvement. In fact,

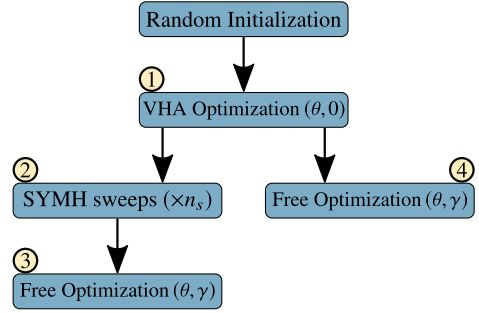


Figure 9: Flow diagram of the optimization schedules employed in the VQE implementation. (1) The parameters in the VHA are optimized respecting the ansatz constrains. (2) We employ SYMH to hop in the landscape. (3)-(4) In a free optimization, the parameters  $\theta$  and  $\gamma$  are independently optimized, hence breaking the constrains in the HVA.

for all values of  $n$  the improvement is larger than 7%.

## 7 Discussion

Analyzing the cost function landscape of variational quantum algorithms and quantum neural networks is a fundamental task to improve their performance. While some rigorous results have been derived that examine the connection between the ansatz and the cost landscape, much still remains to be done. In this work we discussed two phenomena related to the cost function landscape, the first phenomenon being an exponentially large symmetry in the parameters of a PQC, and the second pertains to how quantum noise affects these symmetries. The latter provides the theoretical grounding for the SYMH optimization method. We will now separately summarise these results and discuss their implications.

### 7.1 Exponential symmetry in PQCs

The first phenomenon can be condensed into the following: for buffered PQCs, there are exponentially many sets of symmetric parameters  $\theta$  and  $\theta'$  such that  $V(\theta) = V(\theta')$  (up to a global phase). These symmetries translate in turn into exponential degenerancies in the cost landscape. To understand and analyze these symmetries we have introduced the so-called  $\sigma$ -Pulse method. The main idea behind this method is the creation, propagation, and absorption of virtual gates in the PQC, which allow us to obtain symmetric sets of parameters. Despite the simple interpretation of the  $\sigma$ -Pulse method, its implications are non-trivial. For instance, we can show that all the relevant features of the cost landscape can be found in a subspace of the parameter hyperspace, hence providing an exponential reduction of the search space of variational quantum algorithms.

To the best of our knowledge, the current work represents the first discussion of such a broad degener-

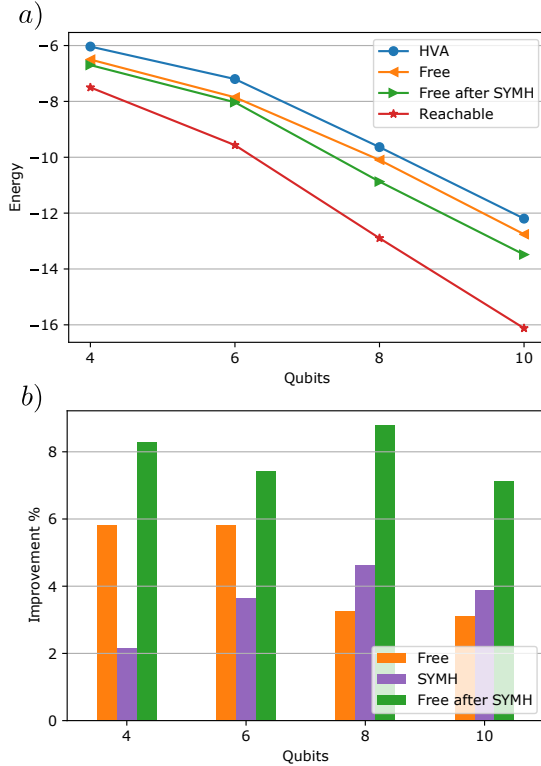


Figure 10: Numerical results for the Variational Quantum Eigensolver implementation. (a) Cost function value (for the best run) versus number of qubits  $n$ . Each curve represents a different optimization scheme as shown in Fig. 9. For all values of  $n$  employing SYMH and breaking the ansatz parameter constrains leads to the best improvement. (b) Cost function value improvement versus number of qubits for different optimization schemes. The improvement is defined in Eq. (22). For all values of  $n$  the largest improvement arises by employing SYMH and breaking the ansatz parameter constrains.

acy in quantum parameterized quantum circuits (al-  
tho subsequent works have studied existence of ex-  
ponentially many local minima of overparameterized  
quantum neural networks [63]). Nonetheless, there  
exists a connection with concepts in Measurement-  
Based Quantum Computing (MBQC) [64], and with  
the concept of flow [65] in particular. The symmetries  
presented here are related to the correction operations  
following a measurement to a qubit. A variant of the  
"angles can be restricted to  $[0, \pi)$ " principle is even  
known for such measurement operations <sup>1</sup>.

In the context of variational algorithms, the result  
is however novel and may have numerous implica-  
tions. For instance, the fact that the noiseless quan-  
tum landscape has been demonstrated to consist of  
an exponentially large number of translated copies of  
a single landscape "unit cell" means that the informa-  
tional content of the landscape is actually greatly re-  
duced. As such, one may hope to construct quantum

<sup>1</sup>Private communication with Will Simmons

algorithms that are more sample-efficient by exploit-  
ing these regularities. Beyond variational algorithms,  
quantum compilers might leverage the symmetries to  
reduce the single-qubit gate count from static circuits.  
Indeed, any  $R_\mu(\pi)$  gate may be eliminated by shifting  
it to an identity gate via an appropriate  $\sigma$ -Pulse.  
This might also be done in an approximate manner, for ex-  
ample if the rotation is only a small  $\epsilon$  away from  $\pi$ .  
However such an approach would leave the circuit's  $T$   
count invariant.

## 7.2 Noise-broken symmetries and SYMH

The second theoretical contribution considers what  
happens when noise is introduced in such a buffered  
PQC. First we rigorously showed that the parameter  
symmetries are preserved under the action of unital  
Pauli noise, implying that dephasing and depolarizing  
noise acting throughout the PQC have no effect on the  
overall symmetric structure of the cost landscape. We  
then proved that non-unital Pauli can break the pa-  
rameter symmetries and hence the degeneracies in  
the cost landscape. This result implies that, when  
training in the presence of noise, some of the previ-  
ously exponentially degenerate global minima can be-  
come local minima. Hence, optimization strategies  
that randomly initialize the parameters could converge  
to one of those local minima and not obtain an opti-  
mal solution.

To mitigate the effect of noise when training in  
degeneracy-broken landscapes, we introduced a novel  
optimization method which we call the Symmetry-  
based Minima Hopping (SYMH) optimizer. SYMH  
employs the parameter symmetries to hop around the  
landscape and attempt to converge to more noise re-  
sistant minima. The main advantage of SYMH is its  
versatility, in the sense that it can be easily com-  
bined with any optimization or error mitigation tech-  
nique without significantly increasing the computa-  
tional overhead.

To showcase the effectiveness of SYMH we numeri-  
cally simulated two variational quantum algorithms  
in the presence of quantum noise. Namely, we imple-  
mented a quantum compiling task and the Variational  
Quantum Eigensolver. In both cases we heuristically  
showed that employing SYMH when optimizing con-  
sistently improves the solution quality. We remark  
that these are preliminary results, and more thorough  
approaches to constructing optimizers with SYMH  
might well prove to be even more fruitful. Hence,  
SYMH is an additional tool in the quantum varia-  
tional toolbox, and it can be regarded as a quantum-  
aware optimizer that accounts for symmetries and  
symmetry breaking in noisy PQCs.

At the same time, one should be mindful that  
SYMH in its present formulation cannot be employed  
as a standalone optimizer, but must be paired with a  
suitable local optimization routine. The local gradient

descent step can be kept minimal, but it is nonetheless unavoidable as without it the optimization would be constrained to a discrete grid of points connected by  $\sigma$ -Pulse symmetries, where it is highly unlikely that the global minimum lies.

To better gauge the utility of SYMH, it is helpful to compare it with similar but non-quantum native approaches to global optimization. The classical machine learning literature offers several such algorithms, the main class being simulated annealing (SA) optimizers [66], which include Metropolis-based algorithms like the particle-collision algorithm [67]. These explore a number of random candidate solutions and select the best ones based on a probabilistic method. The similarity with SYMH is that such algorithms are designed to evade local minima and explore as much of the landscape as possible. Some of them may also incorporate a gradient-based local subroutine [68]. However, one considerable difference is that these algorithms are still local, as at each step they make gradual deviations away from the starting point. Setting a large step size would lead to unpredictable variations in the cost. Instead, SYMH is in principle able to make large jumps across the landscape while limiting the change in cost, as the algorithm is rooted on knowledge of the underlying symmetries of the landscape. Similar considerations also hold for the method of gradient descent with warm restarts [69]. Nonetheless, it is an interesting question for future research if a combination of SA-style approaches and SYMH jumps would be more effective than either for a quantum problem.

Interestingly, the SYMH method also has a natural connection with the broad family of randomized methods for noise mitigation, which include randomized compiling and benchmarking [70, 71], twirling [72] and probabilistic error cancellation [73]. In fact the parameter symmetries obtained through our  $\sigma$ -Pulse method are a special subset of twirling operations that preserve the structure of the ansatz. Nonetheless, the aforementioned methods involve a modification of the circuit via the addition of extra Pauli gates, while we achieve this implicitly by shifting the parameter values. Another difference is that the symmetries are specific to variational circuits, and violate the assumption of Clifford operations required by most twirling-based methods. As noise mitigation techniques, SYMH and randomized error mitigation methods are distinct. The latter aim at converting general noise channels into Pauli ones by a symmetrization procedure, while SYMH leverages the broken symmetries to achieve an improved cost function. In addition SYMH does not require any averaging operation, leading to a smaller overhead.

Despite its merits, SYMH presents some limitations that we now discuss. First, the parameter sweeping algorithm is purely heuristic and comes with no guarantee of convergence. Certainly, if one were given

knowledge of the precise noise channels affecting the system, it is reasonable to expect that more refined optimization methods could be identified. As such, we expect that more advanced versions of SYMH can be derived given knowledge of the noise structure. Still, the numerical results found here suggest that the sweeping method is able to improve the cost function significantly, see Figure 10.

Secondly, even with an ideal optimization schedule SYMH would only be effective up to unital channels, which cannot be corrected by SYMH. These channels, e.g. Pauli channels, are widely present in quantum computers [74]. However, is it also true that many noise mitigation methods exist that are effective for unital Pauli channels [73, 75, 76, 77], and such methods may be easily integrated alongside SYMH. Ultimately we propose SYMH as a first step towards leveraging the quantum landscape structure to optimize cost functions. Moreover, SYMH should not be considered as a standalone technique, but one to be used in conjunction with more standard strategies.

Several future research directions follow for the SYMH method. First, we heuristically verified that hopping does not lead to a critical point, but rather to the valley of a minimum. Analyzing how much the minima can shift could provide additional guarantees for SYMH without a second optimization. Second, in our numerics we observed that shifting some parameters leads to the greatest improvements, but no clear pattern emerged. We leave for future work the analysis of this phenomenon.

## 8 ACKNOWLEDGEMENTS

We thank Kunal Sharma and Lukasz Cincio for helpful discussions. We acknowledge the help of Alexander Cowtan and Ross Duncan through discussions regarding ZX-calculus and its applications to circuit compilation. We are grateful to Will Simmons for pointing out the connection between the parameter symmetries and the concept of flow in MBQC. EF was initially supported by the U.S. Department of Energy (DOE) through a quantum computing program sponsored by the LANL Information Science & Technology Institute. EF and IR acknowledge the support of the UK government department for Business, Energy and Industrial Strategy through the UK national quantum technologies programme. EF acknowledges the support of an industrial CASE (iCASE) studentship, funded by the UK Engineering and Physical Sciences Research Council (grant EP/T517665/1). MC acknowledges initial support from the Center for Non-linear Studies at LANL. AA was initially supported by the Laboratory Directed Research and Development (LDRD) program of LANL under project number 20190065DR. PJC acknowledges initial support from the LANL ASC Beyond Moore’s Law project. This work was supported by the U.S. DOE, Office of

Science, Office of Advanced Scientific Computing Research, under the Quantum Computing Application Teams program.

## References

- [1] J. Preskill. Quantum computing in the NISQ era and beyond. *Quantum*, 2:79, 2018. DOI: 10.22331/q-2018-08-06-79.
- [2] 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(1):625–644, 2021. DOI: 10.1038/s42254-021-00348-9. URL <https://www.nature.com/articles/s42254-021-00348-9>.
- [3] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O’Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5:4213, 2014. DOI: 10.1038/ncomms5213. URL <https://www.nature.com/articles/ncomms5213>.
- [4] Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics*, 18(2):023023, 2016. DOI: 10.1088/1367-2630/18/2/023023. URL <https://iopscience.iop.org/article/10.1088/1367-2630/18/2/023023/meta>.
- [5] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014. DOI: 10.48550/arXiv.1411.4028. URL <https://arxiv.org/abs/1411.4028>.
- [6] J. Romero, J. P. Olson, and A. Aspuru-Guzik. Quantum autoencoders for efficient compression of quantum data. *Quantum Science and Technology*, 2(4):045001, Dec 2017. DOI: 10.1088/2058-9565/aa8072. URL <https://iopscience.iop.org/article/10.1088/2058-9565/aa8072>.
- [7] Sumeet Khatri, Ryan LaRose, Alexander Poremba, Lukasz Cincio, Andrew T. Sornborger, and Patrick J. Coles. Quantum-assisted quantum compiling. *Quantum*, 3:140, May 2019. ISSN 2521-327X. DOI: 10.22331/q-2019-05-13-140. URL <https://doi.org/10.22331/q-2019-05-13-140>.
- [8] R. LaRose, A. Tikku, É. O’Neel-Judy, L. Cincio, and P. J. Coles. Variational quantum state diagonalization. *npj Quantum Information*, 5:1–10, 2018. DOI: 10.1038/s41534-019-0167-6. URL <https://www.nature.com/articles/s41534-019-0167-6>.
- [9] A. Arrasmith, L. Cincio, A. T. Sornborger, W. H. Zurek, and P. J. Coles. Variational consistent histories as a hybrid algorithm for quantum foundations. *Nature communications*, 10(1):3438, 2019. DOI: 10.1038/s41467-019-11417-0. URL <https://www.nature.com/articles/s41467-019-11417-0>.
- [10] M. Cerezo, Alexander Poremba, Lukasz Cincio, and Patrick J Coles. Variational quantum fidelity estimation. *Quantum*, 4:248, 2020. URL <https://quantum-journal.org/papers/q-2020-03-26-248/>.
- [11] Cristina Cirstoiu, Zoe Holmes, Joseph Iosue, Lukasz Cincio, Patrick J Coles, and Andrew Sornborger. Variational fast forwarding for quantum simulation beyond the coherence time. *npj Quantum Information*, 6(1):1–10, 2020. URL <https://www.nature.com/articles/s41534-020-00302-0>.
- [12] Carlos Bravo-Prieto, Ryan LaRose, M. Cerezo, Yigit Subasi, Lukasz Cincio, and Patrick Coles. Variational quantum linear solver. *arXiv preprint arXiv:1909.05820*, 2019. DOI: 10.48550/arXiv.1909.05820. URL <https://arxiv.org/abs/1909.05820>.
- [13] M. Cerezo, Kunal Sharma, Andrew Arrasmith, and Patrick J Coles. Variational quantum state eigensolver. *arXiv preprint arXiv:2004.01372*, 2020. DOI: 10.48550/arXiv.2004.01372. URL <https://arxiv.org/abs/2004.01372>.
- [14] Ivan Rungger, Nathan Fitzpatrick, Honxiang Chen, CH Alderete, Harriett Apel, Alexander Cowtan, Andrew Patterson, D Munoz Ramo, Yingyue Zhu, Nhung Hong Nguyen, et al. Dynamical mean field theory algorithm and experiment on quantum computers. *arXiv preprint arXiv:1910.04735*, 2019. DOI: 10.48550/arXiv.1910.04735. URL <https://arxiv.org/abs/1910.04735>.
- [15] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. The quest for a quantum neural network. *Quantum Information Processing*, 13(11):2567–2586, 2014. DOI: 10.1007/s11128-014-0809-8. URL <https://link.springer.com/article/10.1007/s11128-014-0809-8>.
- [16] Iris Cong, Soonwon Choi, and Mikhail D Lukin. Quantum convolutional neural networks. *Nature Physics*, 15(12):1273–1278, 2019. DOI: 10.1038/s41567-019-0648-8. URL <https://www.nature.com/articles/s41567-019-0648-8>.
- [17] Kerstin Beer, Dmytro Bondarenko, Terry Farrelly, Tobias J Osborne, Robert Salzmann, Daniel Scheiermann, and Ramona Wolf. Training deep quantum neural networks. *Nature Communications*, 11(1):1–6, 2020. DOI: 10.1038/s41467-020-14454-2. URL <https://www.nature.com/articles/s41467-020-14454-2>.
- [18] Guillaume Verdon, Jason Pye, and Michael Broughton. A universal training algorithm for quantum deep learning. *arXiv preprint arXiv:1806.09729*, 2018.

- DOI: 10.48550/arXiv.1806.09729. URL <https://arxiv.org/abs/1806.09729>.
- [19] Andrew Patterson, Hongxiang Chen, Leonard Wossnig, Simone Severini, Dan Browne, and Ivan Rungger. Quantum state discrimination using noisy quantum neural networks. *Physical Review Research*, 3(1):013063, 2021. DOI: 10.1103/PhysRevResearch.3.013063. URL <https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.3.013063>.
- [20] Patrick Huembeli and Alexandre Dauphin. Characterizing the loss landscape of variational quantum circuits. *Quantum Science and Technology*, 6(2):025011, 2021. DOI: 10.1088/2058-9565/abdbc9. URL <https://iopscience.iop.org/article/10.1088/2058-9565/abdbc9>.
- [21] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. Quantum circuit learning. *Phys. Rev. A*, 98(3):032309, 2018. DOI: 10.1103/PhysRevA.98.032309. URL <https://journals.aps.org/prabstract/10.1103/PhysRevA.98.032309>.
- [22] Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. Evaluating analytic gradients on quantum hardware. *Physical Review A*, 99(3):032331, 2019. DOI: 10.1103/PhysRevA.99.032331. URL <https://journals.aps.org/prabstract/10.1103/PhysRevA.99.032331>.
- [23] Kosuke Mitarai and Keisuke Fujii. Methodology for replacing indirect measurements with direct measurements. *Physical Review Research*, 1(1):013006, 2019. DOI: 10.1103/PhysRevResearch.1.013006. URL <https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.1.013006>.
- [24] M. Cerezo and Patrick J Coles. Higher order derivatives of quantum neural networks with barren plateaus. *Quantum Science and Technology*, 6(2):035006, 2021. DOI: 10.1088/2058-9565/abf51a. URL <https://iopscience.iop.org/article/10.1088/2058-9565/abf51a>.
- [25] Andrea Mari, Thomas R. Bromley, and Nathan Killoran. Estimating the gradient and higher-order derivatives on quantum hardware. *Phys. Rev. A*, 103:012405, Jan 2021. DOI: 10.1103/PhysRevA.103.012405. URL <https://link.aps.org/doi/10.1103/PhysRevA.103.012405>.
- [26] Jonas M Kübler, Andrew Arrasmith, Lukasz Cincio, and Patrick J Coles. An adaptive optimizer for measurement-frugal variational algorithms. *Quantum*, 4:263, 2020. DOI: 10.22331/q-2020-05-11-263. URL <https://quantum-journal.org/papers/q-2020-05-11-263/>.
- [27] Ken M Nakanishi, Keisuke Fujii, and Synge Todo. Sequential minimal optimization for quantum-classical hybrid algorithms. *Physical Review Research*, 2(4):043158, 2020. URL <https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.2.043158>.
- [28] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature communications*, 9(1):4812, 2018. DOI: 10.1038/s41467-018-07090-4. URL <https://www.nature.com/articles/s41467-018-07090-4>.
- [29] M. Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature Communications*, 12(1):1–12, 2021. DOI: 10.1038/s41467-021-21728-w. URL <https://www.nature.com/articles/s41467-021-21728-w>.
- [30] Kunal Sharma, M. Cerezo, Lukasz Cincio, and Patrick J Coles. Trainability of dissipative perceptron-based quantum neural networks. *Physical Review Letters*, 128(18):180505, 2022. DOI: 10.1103/PhysRevLett.128.180505.
- [31] Zoë Holmes, Andrew Arrasmith, Bin Yan, Patrick J. Coles, Andreas Albrecht, and Andrew T Sornborger. Barren plateaus preclude learning scramblers. *Physical Review Letters*, 126(19):190501, 2021. DOI: 10.1103/PhysRevLett.126.190501. URL <https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.126.190501>.
- [32] Arthur Pesah, M. Cerezo, Samson Wang, Tyler Volkoff, Andrew T Sornborger, and Patrick J Coles. Absence of barren plateaus in quantum convolutional neural networks. *Physical Review X*, 11(4):041011, 2021. DOI: 10.1103/PhysRevX.11.041011. URL <https://journals.aps.org/prx/abstract/10.1103/PhysRevX.11.041011>.
- [33] Carlos Ortiz Marrero, Mária Kieferová, and Nathan Wiebe. Entanglement-induced barren plateaus. *PRX Quantum*, 2(4):040316, 2021. DOI: 10.1103/PRXQuantum.2.040316.
- [34] 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. DOI: 10.1109/QCE49297.2020.00060. URL <https://ieeexplore.ieee.org/abstract/document/9259938>.
- [35] 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):1–11, 2021. DOI: 10.1038/s41467-021-27045-

6. URL <https://www.nature.com/articles/s41467-021-27045-6>.
- [36] Kunal Sharma, Sumeet Khatri, M. Cerezo, and Patrick J Coles. Noise resilience of variational quantum compiling. *New Journal of Physics*, 22(4):043006, 2020. DOI: 10.1088/1367-2630/ab784c. URL <https://iopscience.iop.org/article/10.1088/1367-2630/ab784c>.
- [37] Enrico Fontana, Nathan Fitzpatrick, David Muñoz Ramo, Ross Duncan, and Ivan Rungger. Evaluating the noise resilience of variational quantum algorithms. *Physical Review A*, 104(2):022403, 2021. DOI: 10.1103/PhysRevA.104.022403. URL <https://journals.aps.org/prabstract/10.1103/PhysRevA.104.022403>.
- [38] James Stokes, Josh Izaac, Nathan Killoran, and Giuseppe Carleo. Quantum natural gradient. *Quantum*, 4:269, 2020. DOI: 10.22331/q-2020-05-25-269. URL <https://quantum-journal.org/papers/q-2020-05-25-269/>.
- [39] Bálint Koczor and Simon C Benjamin. Quantum natural gradient generalised to non-unitary circuits. *arXiv preprint arXiv:1912.08660*, 2019. DOI: 10.48550/arXiv.1912.08660. URL <https://arxiv.org/abs/1912.08660>.
- [40] Ken M Nakanishi, Keisuke Fujii, and Syngae Todo. Sequential minimal optimization for quantum-classical hybrid algorithms. *Physical Review Research*, 2(4):043158, 2020. DOI: 10.1103/PhysRevResearch.2.043158. URL <https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.2.043158>.
- [41] Andrew Arrasmith, Lukasz Cincio, Rolando D Somma, and Patrick J Coles. Operator sampling for shot-frugal optimization in variational algorithms. *arXiv preprint arXiv:2004.06252*, 2020. DOI: 10.48550/arXiv.2004.06252. URL <https://arxiv.org/abs/2004.06252>.
- [42] Ryan Sweke, Frederik Wilde, Johannes Jakob Meyer, Maria Schuld, Paul K Fährmann, Barthélémy Meynard-Piganeau, and Jens Eisert. Stochastic gradient descent for hybrid quantum-classical optimization. *Quantum*, 4:314, 2020. DOI: 10.22331/q-2020-08-31-314. URL <https://quantum-journal.org/papers/q-2020-08-31-314/>.
- [43] Kevin J Sung, Jiahao Yao, Matthew P Harrigan, Nicholas C Rubin, Zhang Jiang, Lin Lin, Ryan Babbush, and Jarrod R McClean. Using models to improve optimizers for variational quantum algorithms. *Quantum Science and Technology*, 5(4):044008, 2020. DOI: 10.1088/2058-9565/abb6d9. URL <https://iopscience.iop.org/article/10.1088/2058-9565/abb6d9>.
- [44] Wim Lavrijsen, Ana Tudor, Juliane Müller, Costin Iancu, and Wibe de Jong. Classical optimizers for noisy intermediate-scale quantum devices. *arXiv preprint arXiv:2004.03004*, 2020. DOI: 10.1109/QCE49297.2020.00041. URL <https://arxiv.org/abs/2004.03004>.
- [45] Aram Harrow and John Napp. Low-depth gradient measurements can improve convergence in variational hybrid quantum-classical algorithms. *arXiv preprint arXiv:1901.05374*, 2019. URL <https://arxiv.org/abs/1901.05374>.
- [46] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242, 2017. DOI: 10.1038/nature23879. URL <https://www.nature.com/articles/nature23879>.
- [47] S. Hadfield, Z. Wang, B. O’Gorman, E. G. Rieffel, D. Venturelli, and R. Biswas. From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms*, 12(2):34, Feb 2019. ISSN 1999-4893. DOI: 10.3390/a12020034. URL <https://www.mdpi.com/1999-4893/12/2/34>.
- [48] Yudong Cao, Jonathan Romero, Jonathan P Olson, Matthias Degroote, Peter D Johnson, Mária Kieferová, Ian D Kivlichan, Tim Menke, Borja Peropadre, Nicolas PD Sawaya, et al. Quantum chemistry in the age of quantum computing. *Chemical reviews*, 119(19):10856–10915, 2019. DOI: 10.1021/acs.chemrev.8b00803. URL <https://pubs.acs.org/doi/10.1021/acs.chemrev.8b00803>.
- [49] Rodney J Bartlett and Monika Musiał. Coupled-cluster theory in quantum chemistry. *Reviews of Modern Physics*, 79(1):291, 2007. DOI: 10.1103/RevModPhys.79.291. URL <https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.79.291>.
- [50] Joonho Lee, William J Huggins, Martin Head-Gordon, and K Birgitta Whaley. Generalized unitary coupled cluster wave functions for quantum computation. *Journal of chemical theory and computation*, 15(1):311–324, 2018. DOI: 10.1021/acs.jctc.8b01004. URL <https://pubs.acs.org/doi/10.1021/acs.jctc.8b01004>.
- [51] Bob Coecke and Ross Duncan. Interacting quantum observables: categorical algebra and diagrammatics. *New Journal of Physics*, 13(4):043016, 2011. DOI: 10.1088/1367-2630/13/4/043016. URL <https://iopscience.iop.org/article/10.1088/1367-2630/13/4/043016>.
- [52] Daniel Stilck França and Raul Garcia-Patron. Limitations of optimization algorithms on noisy quantum devices. *Nature Physics*, 17(11):1221–1227, 2021. DOI: 10.1038/s41567-021-01356-3. URL <https://www.nature.com/articles/s41567-021-01356-3>.



- [53] Bryan T Gard, Linghua Zhu, George S Barron, Nicholas J Mayhall, Sophia E Economou, and Edwin Barnes. Efficient symmetry-preserving state preparation circuits for the variational quantum eigensolver algorithm. *npj Quantum Information*, 6(1):1–9, 2020. DOI: [10.1038/s41534-019-0240-1](https://doi.org/10.1038/s41534-019-0240-1). URL <https://www.nature.com/articles/s41534-019-0240-1>.
- [54] Michael Streif, Martin Leib, Filip Wudarski, Eleanor Rieffel, and Zihui Wang. Quantum algorithms with local particle-number conservation: Noise effects and error correction. *Physical Review A*, 103(4):042412, 2021. DOI: [10.1103/PhysRevA.103.042412](https://doi.org/10.1103/PhysRevA.103.042412). URL <https://journals.aps.org/prabstract/10.1103/PhysRevA.103.042412>.
- [55] F. T. Chong, D. Franklin, and M. Martonosi. Programming languages and compiler design for realistic quantum hardware. *Nature*, 549(7671):180, 2017. DOI: [10.1038/nature23459](https://doi.org/10.1038/nature23459). URL <https://www.nature.com/articles/nature23459>.
- [56] Thomas Häner, Damian S Steiger, Krysta Svore, and Matthias Troyer. A software methodology for compiling quantum programs. *Quantum Science and Technology*, 3(2):020501, 2018. DOI: [10.1088/2058-9565/aaa5cc](https://doi.org/10.1088/2058-9565/aaa5cc). URL <https://iopscience.iop.org/article/10.1088/2058-9565/aaa5cc>.
- [57] D. Venturelli, M. Do, E. Rieffel, and J. Frank. Compiling quantum circuits to realistic hardware architectures using temporal planners. *Quantum Science and Technology*, 3(2):025004, 2018. DOI: [10.1088/2058-9565/aaa331](https://doi.org/10.1088/2058-9565/aaa331). URL <https://iopscience.iop.org/article/10.1088/2058-9565/aaa331>.
- [58] Tyson Jones and Simon C Benjamin. Robust quantum compilation and circuit optimisation via energy minimisation. *Quantum*, 6:628, 2022. DOI: [10.22331/q-2022-01-24-628](https://doi.org/10.22331/q-2022-01-24-628). URL <https://quantum-journal.org/papers/q-2022-01-24-628/>.
- [59] Kentaro Heya, Yasunari Suzuki, Yasunobu Nakamura, and Keisuke Fujii. Variational quantum gate optimization. *arXiv preprint arXiv:1810.12745*, 2018. DOI: [10.48550/arXiv.1810.12745](https://doi.org/10.48550/arXiv.1810.12745). URL <https://arxiv.org/abs/1810.12745>.
- [60] M. J. D. Powell. The BOBYQA algorithm for bound constrained optimization without derivatives. *Technical Report, Department of Applied Mathematics and Theoretical Physics*, 01 2009. URL [https://www.damtp.cam.ac.uk/user/na/NA\\_papers/NA2009\\_06.pdf](https://www.damtp.cam.ac.uk/user/na/NA_papers/NA2009_06.pdf).
- [61] Dave Wecker, Matthew B Hastings, and Matthias Troyer. Progress towards practical quantum variational algorithms. *Physical Review A*, 92(4):042303, 2015. DOI: [10.1103/PhysRevA.92.042303](https://doi.org/10.1103/PhysRevA.92.042303). URL <https://journals.aps.org/prabstract/10.1103/PhysRevA.92.042303>.
- [62] Roeland Wiersema, Cunlu Zhou, Yvette de Sereville, Juan Felipe Carrasquilla, Yong Baek Kim, and Henry Yuen. Exploring entanglement and optimization within the hamiltonian variational ansatz. *PRX Quantum*, 1(2):020319, 2020. DOI: [10.1103/PRXQuantum.1.020319](https://doi.org/10.1103/PRXQuantum.1.020319). URL <https://journals.aps.org/prxquantum/pdf/10.1103/PRXQuantum.1.020319>.
- [63] Xuchen You and Xiaodi Wu. Exponentially many local minima in quantum neural networks. In *International Conference on Machine Learning*, pages 12144–12155. PMLR, 2021. URL <https://proceedings.mlr.press/v139/you21c.html>.
- [64] Hans J Briegel, David E Browne, Wolfgang Dür, Robert Raussendorf, and Maarten Van den Nest. Measurement-based quantum computation. *Nature Physics*, 5(1):19–26, 2009. DOI: [10.1038/nphys1157](https://doi.org/10.1038/nphys1157). URL <https://www.nature.com/articles/nphys1157>.
- [65] Vincent Danos and Elham Kashefi. Determinism in the one-way model. *Physical Review A*, 74(5):052310, 2006. DOI: [10.1103/PhysRevA.74.052310](https://doi.org/10.1103/PhysRevA.74.052310). URL <https://journals.aps.org/prabstract/10.1103/PhysRevA.74.052310>.
- [66] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983. DOI: [10.1126/science.220.4598.671](https://doi.org/10.1126/science.220.4598.671). URL <https://www.science.org/doi/abs/10.1126/science.220.4598.671>.
- [67] Wagner F Sacco and CREA Oliveira. A new stochastic optimization algorithm based on a particle collision metaheuristic. *Proceedings of 6th WCSMO*, 2005. URL <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.80.6308&rep=rep1&type=pdf>.
- [68] Ana Carolina Rios-Coelho, Wagner F Sacco, and Nélio Henderson. A metropolis algorithm combined with hooke-jeeves local search method applied to global optimization. *Applied Mathematics and Computation*, 217(2):843–853, 2010. DOI: [10.1016/j.amc.2010.06.027](https://doi.org/10.1016/j.amc.2010.06.027). URL <https://www.sciencedirect.com/science/article/pii/S0096300310007125>.
- [69] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. DOI: [10.48550/arXiv.1608.03983](https://doi.org/10.48550/arXiv.1608.03983). URL <https://arxiv.org/abs/1608.03983>.
- [70] Oliver Kern, Gernot Alber, and Dima L Shepelyansky. Quantum error correction of coherent errors by randomization. *The European Physical Journal D-Atomic, Molec-*

- ular, *Optical and Plasma Physics*, 32(1): 153–156, 2005. DOI: 10.1140/epjd/e2004-00196-9. URL <https://link.springer.com/article/10.1140/epjd/e2004-00196-9>.
- [71] Joel J Wallman and Joseph Emerson. Noise tailoring for scalable quantum computation via randomized compiling. *Physical Review A*, 94(5):052325, 2016. URL <https://journals.aps.org/pr/abstract/10.1103/PhysRevA.94.052325>.
- [72] Osama Moussa, Marcus P da Silva, Colm A Ryan, and Raymond Laflamme. Practical experimental certification of computational quantum gates using a twirling procedure. *Physical review letters*, 109(7):070504, 2012. DOI: 10.1103/PhysRevLett.109.070504. URL <https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.109.070504>.
- [73] Kristan Temme, Sergey Bravyi, and Jay M Gambetta. Error mitigation for short-depth quantum circuits. *Physical review letters*, 119(18):180509, 2017. DOI: 10.1103/PhysRevLett.119.180509. URL <https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.119.180509>.
- [74] Steven T Flammia and Joel J Wallman. Efficient estimation of pauli channels. *ACM Transactions on Quantum Computing*, 1(1):1–32, 2020. DOI: 10.1145/3408039. URL <https://dl.acm.org/doi/abs/10.1145/3408039>.
- [75] Ying Li and Simon C Benjamin. Efficient variational quantum simulator incorporating active error minimization. *Physical Review X*, 7(2):021050, 2017. URL <https://journals.aps.org/prx/abstract/10.1103/PhysRevX.7.021050>.
- [76] Suguru Endo, Simon C Benjamin, and Ying Li. Practical quantum error mitigation for near-future applications. *Physical Review X*, 8(3):031027, 2018. DOI: 10.1103/PhysRevX.8.031027. URL <https://journals.aps.org/prx/abstract/10.1103/PhysRevX.8.031027>.
- [77] 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. *Physical Review Letters*, 127(27):270502, 2021. DOI: 10.1103/PhysRevLett.127.270502. URL <https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.127.270502>.

## APPENDIX

In this appendix we provide of our main results, with Appendix A containing the proof of Proposition 1 and Corollary 1. Then, in Appendix B we present the proofs of Theorem 1 and Theorem 2.

### A Proof of Proposition 1 and Corollary 1

In what follows we present the proof for Proposition 1.

*Proof.* Let us first recall that, as mentioned in the main text, there are  $M$  parameters in the PQC  $V(\boldsymbol{\theta})$ . Moreover let us denote as  $\mathcal{G}$  the set of all possible generator choices. The number of distinct symmetric sets of parameters  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  obtained through the  $\sigma$ -Pulse method is given by the cardinality of  $\mathcal{G}$ :

$$|\mathcal{G}| = \sum_{\alpha=0}^M \binom{M}{\alpha} = 2^M. \quad (23)$$

Similarly, we can count the number of symmetric sets of parameters as the number of bitstrings  $\boldsymbol{\beta}$  in  $\mathcal{B}$  of length  $M$ , which is precisely  $2^M$ .  $\square$

Now we prove Corollary 1. In particular we show that given any  $\{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$  one can always find a set  $\{\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\gamma}}\}$  with  $\boldsymbol{\beta} = \mathbf{0}$ .

*Proof.* Let us assume that the parameters in  $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots)$  are order by layer, where a layer consists of quantum gates that can be performed in parallel and where the first layer contains the first gates in  $V(\boldsymbol{\theta})$ . In general we can assume without loss of generality that the angles in  $\boldsymbol{\theta}$  are in  $[0, 2\pi)$ . We now describe a sequential procedure that can be used to obtain the vector  $\tilde{\boldsymbol{\theta}}$  where every parameter  $\tilde{\theta}_j \in \tilde{\boldsymbol{\theta}}$  not in the buffer layer are in the reduced domain  $[0, \pi)$ .

If  $\theta_1 \in [0, \pi)$  then we do nothing, but if  $\theta_1 \in [\pi, 2\pi)$  we create and forward propagate a  $\sigma$ -Pulse. According to Eq. (7), this will add  $\pi$  to  $\theta_1$ , which maps it to the interval  $[0, \pi)$ . This procedure is then sequentially repeated for each parameter in  $\boldsymbol{\theta}$  not in the buffer layer. We remark that since  $\sigma$ -Pulse propagate forward in the circuit, creating a pulse in  $\theta_j$  does not affect any angle  $\theta_k$  with  $k < j$ . Moreover, we know from (9) that as the  $\sigma$ -Pulses propagate they can add a minus sign to other angles in  $\boldsymbol{\theta}$ . Hence, if a given  $\theta_j$  that was originally in  $[\pi, 2\pi)$  picked up a minus sign then we do nothing as it will now be in  $[0, \pi)$ . On the other hand, if it was in  $[0, \pi)$  we have to create a  $\sigma$ -Pulse to map it to  $[0, \pi)$ . Note that at the end of this procedure every parameter not in the buffer layer will be mapped to the reduced domain  $[0, \pi)$ .  $\square$

## B Proof of our main theorems

### B.1 Proof of Theorem 1

Let us start by presenting a more detailed definition of a unital Pauli channel:

**Unital Pauli channels.** A Pauli noise channel corresponds to the action of random Pauli operators according to a given probability distribution. Let  $\mathcal{P}_U$  denote an  $n$ -qubit Pauli channel. The action of  $\mathcal{P}_U$  on any given  $n$ -qubit Pauli operator is given by

$$\mathcal{P}_U(X^a Z^b) = \sum_{l,k} p_{l,k} X^l Z^k (X^a Z^b) (X^l Z^k)^\dagger, \quad (24)$$

where  $0 \leq p_{l,k}^A \leq 1$ , and  $\sum_{l,k} p_{l,k}^A = 1$ . By using the fact that

$$X^l Z^k = (-1)^{l \cdot k} Z^k X^l, \quad (25)$$

we find

$$\begin{aligned} \mathcal{P}_U(X^a Z^b) &= \sum_{l,k} p_{l,k} X^l Z^k X^a Z^b Z^k X^l \\ &= \sum_{l,k} (-1)^{a \cdot k} (-1)^{b \cdot l} p_{l,k} X^a Z^b \\ &= p_{\mathbf{a}, \mathbf{b}} X^a Z^b, \end{aligned} \quad (26)$$

where  $p_{\mathbf{a}, \mathbf{b}} = \sum_{l,k} (-1)^{a \cdot k} (-1)^{b \cdot l} p_{l,k}$  and  $-1 \leq p_{\mathbf{a}, \mathbf{b}} \leq 1$  for all  $\mathbf{a}, \mathbf{b} \in \{0, 1\}^n$ .

We now prove Theorem 1.

*Proof.* Let us now consider a buffered circuit  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  which is implemented in the presence of a unital Pauli noise mode as presented in Definition 4. Here we show that propagating the pulses (Pauli operators) through the circuit does not change the unitary being produced as the Pauli operators commute with unital Pauli noise.

Let us now consider the channel  $\mathcal{V}_B$  which implements the unitary  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$ . This channel can be expressed as

$$\mathcal{V}_B = \mathcal{U}_B \circ \mathcal{V}_L \circ \dots \circ \mathcal{V}_1, \quad (27)$$

where  $\mathcal{V}_l$  is the channel that implements the unitaries in the  $l$ -th layer, and where  $\mathcal{U}_B$  the channel that implements the buffer unitary. From the  $\sigma$ -Pulse method we know that we can find a symmetric set of parameters by creating a sigma pulse, propagating it, and absorbing it in the buffer layer. In the channel notation this procedure can be expressed as

- Creation of a primary  $\sigma$ -Pulse:

$$\mathcal{V}_B = \mathcal{U}_B \circ \mathcal{V}_L \circ \dots \circ \Sigma_1 \circ \tilde{\mathcal{V}}_1, \quad (28)$$

with  $\Sigma_1$  the channel that implements the  $\sigma$ -Pulse, and with  $\tilde{\mathcal{V}}_1$  the parameter shifted unitary.

- Propagation of the  $\sigma$ -Pulses:

$$\mathcal{V}_B = \mathcal{U}_B \circ \Sigma_L \circ \tilde{\mathcal{V}}_L \circ \cdots \circ \tilde{\mathcal{V}}_1, \quad (29)$$

where now  $\Sigma_L$  is the channel that implements the primary and secondary  $\sigma$ -Pulses.

- Absorption of the  $\sigma$ -Pulses:

$$\mathcal{V}_B = \tilde{\mathcal{U}}_B \circ \tilde{\mathcal{V}}_L \circ \cdots \circ \tilde{\mathcal{V}}_1. \quad (30)$$

Here we can see that the channel  $\mathcal{V}_B$  remains unchanged, meaning that  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma}) = V(\boldsymbol{\theta}, \tilde{\boldsymbol{\gamma}})$ .

Let us now analyze this procedure in the presence of noise. The noisy version of the channel that implements  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  can be expressed as

$$\widehat{\mathcal{V}}_B = \mathcal{P}_U^{(L+2)} \circ \mathcal{U}_B \circ \mathcal{P}_U^{(L+1)} \circ \mathcal{V}_L \circ \mathcal{P}_U^{(L)} \cdots \circ \mathcal{P}_U^{(1)} \circ \mathcal{V}_1 \circ \mathcal{P}_U^{(0)}, \quad (31)$$

with  $\mathcal{P}_U^{(l)}$  the noisy channel acting after every layer of gates. Once a  $\sigma$ -Pulse has been created we will have

$$\widehat{\mathcal{V}}_B = \mathcal{P}_U^{(L+2)} \circ \mathcal{U}_B \circ \mathcal{P}_U^{(L+1)} \circ \mathcal{V}_L \circ \mathcal{P}_U^{(L)} \cdots \circ \mathcal{P}_U^{(1)} \circ \Sigma_1 \circ \tilde{\mathcal{V}}_1 \circ \mathcal{P}_U^{(0)}. \quad (32)$$

As we now show, any unital noisy channels  $\mathcal{P}_U^{(i)}$  always commute with the channels  $\Sigma_k$  that implement  $\sigma$ -Pulses. Explicitly, the action of on any given  $n$ -qubit Pauli operator is given by

$$\begin{aligned} \Sigma_k(X^a Z^b) &= X^p Z^q X^a Z^b (X^p Z^q)^\dagger \\ &= (-1)^{a \cdot q} (-1)^{b \cdot p} X^a Z^b. \end{aligned} \quad (33)$$

Hence, using Eqs. (25), (33) and (24) we have that the following chain of equalities always hold

$$\begin{aligned} \mathcal{P}_U^{(i)} \circ \Sigma_k(X^a Z^b) &= \sum_{l,k} p_{l,k} X^l Z^k X^p Z^q (X^a Z^b) Z^q X^p Z^k X^l \\ &= \sum_{l,k} p_{l,k} X^l X^p Z^k Z^q (X^a Z^b) Z^q Z^k X^p X^l \\ &= \sum_{l,k} p_{l,k} X^{l+p} Z^{k+q} (X^a Z^b) Z^{q+k} X^{p+l} \\ &= \Sigma_k \circ \mathcal{P}_U^{(i)}(X^a Z^b). \end{aligned} \quad (34)$$

From 34 we can commute the  $\sigma$ -Pulses through the noisy channel so that

$$\begin{aligned} \widehat{\mathcal{V}}_B &= \mathcal{P}_U^{(L+2)} \circ \mathcal{U}_B \circ \mathcal{P}_U^{(L+1)} \circ \mathcal{V}_L \circ \mathcal{P}_U^{(L)} \cdots \circ \mathcal{P}_U^{(1)} \circ \Sigma_1 \circ \mathcal{V}_1 \circ \mathcal{P}_U^{(0)} \\ &= \mathcal{P}_U^{(L+2)} \circ \mathcal{U}_B \circ \Sigma_L \circ \mathcal{P}_U^{(L+1)} \circ \tilde{\mathcal{V}}_L \circ \mathcal{P}_U^{(L)} \cdots \circ \mathcal{P}_U^{(1)} \circ \tilde{\mathcal{V}}_1 \circ \mathcal{P}_U^{(0)} \\ &= \mathcal{P}_U^{(L+2)} \circ \tilde{\mathcal{U}}_B \circ \mathcal{P}_U^{(L+1)} \circ \tilde{\mathcal{V}}_L \circ \mathcal{P}_U^{(L)} \cdots \circ \mathcal{P}_U^{(1)} \circ \tilde{\mathcal{V}}_1 \circ \mathcal{P}_U^{(0)}. \end{aligned} \quad (35)$$

Equation (35) proves Theorem 1 as it shows that the noisy channels that implement  $V(\boldsymbol{\theta}, \boldsymbol{\gamma})$  and  $V(\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\gamma}})$  are the same.  $\square$

## B.2 Proof of Theorem 2

We now prove Theorem 2. Here we show that there exist noisy channels  $\mathcal{N}$  (which include as special case

then non-unital Pauli noise channels  $\mathcal{P}_{NU}$ ) that do not commute with the channels  $\Sigma$  that implement the  $\sigma$ -Pulses and which hence break the parameter symmetries.

*Proof.* Let us consider a general noisy channel which acts on an  $n$ -qubit bitstring as

$$\mathcal{N}(X^a Z^b) = \sum_{k,l} c_{k,l}^{a,b} X^k Z^l, \quad (36)$$

which has some  $c_{k,l}^{a,b} \neq 0$  for  $k, l \neq a, b$ . A non-unital Pauli channel in particular must have  $c_{k,l}^{0,0} \neq 0$  for some  $k, l \neq 0, 0$  (see Eq. 15 of the main text). Similarly to the previous section, we now can define noisy version of the channel that implements  $V_B(\boldsymbol{\theta}, \boldsymbol{\gamma})$  as

$$\widehat{\mathcal{V}}_B = \mathcal{N}^{(L+2)} \circ \mathcal{U}_B \circ \mathcal{N}^{(L+1)} \circ \mathcal{V}_L \circ \mathcal{N}^{(L)} \cdots \circ \mathcal{N}^{(1)} \circ \mathcal{V}_1 \circ \mathcal{N}^{(0)}, \quad (37)$$

Let us now analyze if the channel  $\Sigma_k$  commutes with a noise channel  $\mathcal{N}$  when acting on an  $n$ -qubit bitstring  $X^a Z^b$ .

$$\begin{aligned} [\Sigma_k, \mathcal{N}](X^a Z^b) &= (\Sigma_k \circ \mathcal{N} - \mathcal{N} \circ \Sigma_k)(X^a Z^b) \\ &= (-1)^{a \cdot q} (-1)^{b \cdot p} \sum_{k,l} c_{k,l}^{a,b} X^k Z^l - \sum_{k,l} c_{k,l}^{a,b} (-1)^{k \cdot q} (-1)^{l \cdot p} X^k Z^l \\ &= \sum_{k,l} [(-1)^{a \cdot q} (-1)^{b \cdot p} c_{k,l}^{a,b} - (-1)^{k \cdot q} (-1)^{l \cdot p} c_{k,l}^{a,b}] X^k Z^l \\ &= \sum_{k,l} [(-1)^{a \cdot q + b \cdot p} - (-1)^{k \cdot q + l \cdot p}] c_{k,l}^{a,b} X^k Z^l \\ &= \sum_{k,l} M_{p,q}^{a,b,k,l} c_{k,l}^{a,b} X^k Z^l \end{aligned} \quad (38)$$

where

$$M_{p,q}^{a,b,k,l} = \begin{cases} 2 & \text{if } a \cdot q = b \cdot p \pmod{2} \text{ and } k \cdot q \neq l \cdot p \pmod{2}, \\ -2 & \text{if } a \cdot q \neq b \cdot p \pmod{2} \text{ and } k \cdot q = l \cdot p \pmod{2}, \\ 0 & \text{otherwise.} \end{cases} \quad (39)$$

Thus, given  $p, q$  and  $a, b$ , there exists noise channel defined by some  $c_{k,l}^{a,b}$  such that the  $\sigma$ -Pulse does not commute with the noise channel. That any such noise channel must be non-Pauli is clear since  $M_{p,q}^{a,b,k,l} = 0$  for all  $p, q$  whenever  $a, b = k, l$ .  $\square$