

# Quantum Policy Gradient Algorithm with Optimized Action Decoding

Nico Meyer,<sup>1,2,\*</sup> Daniel D. Scherer,<sup>1</sup> Axel Plinge,<sup>1</sup> Christopher Mutschler,<sup>1</sup> and Michael J. Hartmann<sup>2</sup>

<sup>1</sup>Fraunhofer IIS, Fraunhofer Institute for Integrated Circuits IIS, Nürnberg, Germany

<sup>2</sup>Friedrich-Alexander University Erlangen-Nürnberg (FAU), Department of Physics, Erlangen, Germany

Quantum machine learning implemented by variational quantum circuits (VQCs) is considered a promising concept for the noisy intermediate-scale quantum computing era. Focusing on applications in quantum reinforcement learning, we propose a specific action decoding procedure for a quantum policy gradient approach. We introduce a novel quality measure that enables us to optimize the classical post-processing required for action selection, inspired by local and global quantum measurements. The resulting algorithm demonstrates a significant performance improvement in several benchmark environments. With this technique, we successfully execute a full training routine on a 5-qubit hardware device. Our method introduces only negligible classical overhead and has the potential to improve VQC-based algorithms beyond the field of quantum reinforcement learning.

## I. INTRODUCTION

Reinforcement learning (RL) currently receives increasing attention due to its potential in a multitude of applications. In an RL setup, an agent aims to learn a control strategy, i.e., a policy, for a specific problem. Training such a policy can require approximating a complex, multimodal distribution, which is often done with a deep neural network (DNN). With increasing problem difficulty, this approach has some caveats, such as an undesirable sampling and model complexity [1–3]. Training data is obtained by step-wise interaction with the environment via actions, which returns a reward value and a new state. Using these samples, one can optimize the parameters of the policy to maximize the long-term reward with gradient-based techniques, thus forming a policy gradient (PG) algorithm. The possibilities of employing RL for real-world problems are abundant – examples can be found in the domains of self-driving cars [4] or MIMO beamforming for mobile communication [5].

Exploring the possibilities offered by other computing paradigms might elevate the impact of RL, e.g. by circumventing the problems caused by increasing parameter complexity of DNN-based machine learning models. Quantum computing (QC), based on the idea of exploiting quantum mechanical properties for computation, might offer advantages in the approximation and sampling from complex distributions. Although the development of quantum computers is still in its infancy, a number of studies have already claimed experimental results beyond classical capabilities on specific tasks [6–8].

The nowadays available quantum devices are considered noisy intermediate-scale quantum (NISQ) hardware, i.e., they only provide a limited amount of qubits that are still significantly affected by noise. Therefore, a major part of current research focuses on strategies and algorithms that are able to cope with these restrictions, while at the same time aiming for computational power beyond what is possible with classical devices. A promising idea

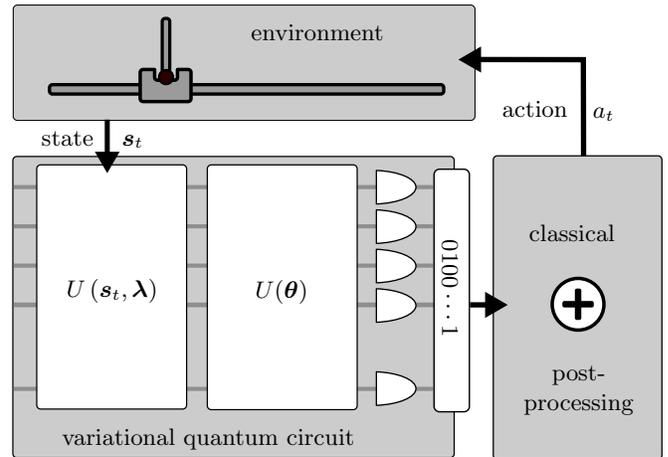


FIG. 1. Proposed method: The prepared quantum state is measured in the computational basis. The results are post-processed (with a function maximizing our proposed *globality measure*) before selecting an action. The update of the parameters  $\Theta = (\theta, \lambda)$  utilizes the same post-processing scheme.

suggests using *variational quantum algorithms* as a platform for quantum machine learning (QML) [9–11], for which a certain degree of resilience to the inevitable hardware noise has been reported [12–15]. Variational quantum algorithms consist of a *variational quantum circuit (VQC)*, which incorporates trainable parameters, and a classical optimization routine to optimize these parameters. When viewing the VQCs as function approximators, the property of universal function approximation holds under certain conditions [16, 17]. For specific problems, variational quantum algorithms are known to exhibit provable quantum advantage [18, 19].

**Contribution.** In variational quantum algorithms, it is necessary to extract classical information from the quantum state prepared by the VQC. Whereas there has been work on obtaining a maximal amount of information about the prepared state via a limited amount of measurements [20], our goal is to group measured bitstrings from a readout in the computational basis, such that a well performing RL strategy emerges. Those aspects have, to our knowledge, not yet been explored

\* nico.meyer@iis.fraunhofer.de

for the framework of VQC-based quantum policy gradient (QPG) algorithms. For the quantum reinforcement learning (QRL) setup, we refer to this task as *action decoding*. Motivated by the RAW-VQC policy (see Sec. III) proposed by Jerbi et al. [21] in Sec. IV, we start with a formulation in terms of projective measurements in Subsec. IV A. This is then decomposed into a measurement in the computational basis and the successive application of a classical post-processing function in Subsec. IV B. Our developed *globality measure* allows to compare specific instances of those functions. As the globality values are shown to strongly correlate with RL performance, we propose a routine to construct an optimal (w.r.t. the globality measure) post-processing function. It is important to mention that, in contrast to the approach by Jerbi et al. [21], our procedure is feasible for problems with large action spaces.

The experimental results in Sec. V support our claims. Empirically, we observe a strong correlation between RL performance and our globality measure in Subsec. V A in the RL environments `CartPole`, `FrozenLake`, and different configurations of `ContextualBandits`. More concretely, training converges much faster (or even at all) for policies with an underlying post-processing function that has a large globality value. The results are supported by an analysis of the effective dimension and Fisher information spectrum in Subsec. V B. As our technique only marginally increases the classical overhead (while reducing the required VQC size) it suggests itself as a tool for execution on NISQ devices. To demonstrate the efficiency of our algorithm, we execute the full RL training routine for a `ContextualBandits` problem on a 5-qubit quantum hardware device in Subsec. V C.

## II. RELATED WORK

A summary of the current body of work on QRL, including also post-NISQ approaches, can be found in the survey by Meyer et al. [22]. Specific QRL routines have already been realized experimentally [23]. An early instance of VQC-based QRL proposes to use a VQC as a function approximator for the action-value function [24]. We follow the lines of Jerbi et al. [21], which uses the VQC for policy approximation, forming a QPG algorithm. Independent work on the QPG approach includes an extension to quantum environments [25], which requires a fundamentally different action decoding strategy. The ideas of value-function and policy approximation are combined into actor-critic approaches [26, 27], which potentially benefit from our contribution as well.

While the algorithmic routine of QPG follows the idea of classical PG, the design of the VQC function approximator is an important ongoing research field. The typical architecture features three different building blocks, i.e., a data encoding layer, (potentially multiple) variational layers with trainable parameters, and some measurement observables that extract information from the prepared

quantum state. There are some guidelines for designing data encoding [17, 28, 29] and variational layers [30, 31], based on the specific problem type. We focus on the necessary measurements, which require special attention in the context of quantum information theory [32, 33], and also QML with VQCs [10, 34, 35]. However, the question how to best measure VQC outputs and classically post-process them to optimize QRL performance, yet alone QPG performance, is still open.

## III. QUANTUM POLICY GRADIENT ALGORITHM

RL is an algorithmic concept to solve a complex task, where data is generated by interaction of an agent with an environment. The setup is usually described as a Markov Decision Process (MDP), i.e. a 5-tuple  $(\mathcal{S}, \mathcal{A}, R, T, \gamma)$ , where  $\mathcal{S}$  is the state set,  $\mathcal{A}$  is the set of available actions,  $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$  is a reward function,  $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$  describes the environment dynamics, and  $0 \leq \gamma \leq 1$  is a discount factor. At each timestep  $t$ , the agent observes the environment state  $\mathbf{s}_t$ , and decides on an action  $a_t$ . This decision is sampled from the current policy  $\pi : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$ , which defines a probability density function (PDF) over all possible actions  $a$  for a given state  $\mathbf{s}$ . The selected action is executed, and the agent receives a scalar reward  $r_t = R(\mathbf{s}_t, a_t)$ , after which the environment transitions to state  $\mathbf{s}_{t+1}$ , following its dynamics  $T$ . The objective is to learn a policy, which maximizes the (discounted) expected return  $G_t := \sum_{t'=t}^{H-1} \gamma^{t'-t} \cdot r_{t'}$  for some horizon  $H < \infty$  [36].

Our work follows the hybrid QPG algorithm proposed by Jerbi et al. [21]. The approach is inspired by the classical REINFORCE idea with function approximation [37], also referred to as *vanilla policy gradient*. Its centerpiece is the parameterized policy  $\pi_{\Theta} : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$ , where  $\Theta$  denotes the trainable parameters of the function approximator. The parameters are updated with a gradient ascent technique, i.e.,  $\Theta \leftarrow \Theta + \alpha \cdot \nabla_{\Theta} J(\Theta)$ , with learning rate  $\alpha$ , and some scalar performance measure  $J(\Theta)$ . The *policy gradient theorem* [37] states the gradient of the performance measure as

$$\nabla_{\Theta} J(\Theta) = \mathbb{E}_{\pi_{\Theta}} \left[ \sum_{t=0}^{H-1} \nabla_{\Theta} \ln \pi_{\Theta}(a_t | \mathbf{s}_t) \cdot (G_t - b(\mathbf{s}_t)) \right], \quad (1)$$

where  $b(\cdot)$  is some baseline function, which is typically chosen to reduce the variance during training. In practice, the expectation value in Eq. (1) is approximated by averaging over several trajectories  $\tau$  (i.e., sequences of current state, executed action, and received reward for several timesteps), that are generated by following the current policy  $\pi_{\Theta}$ . For a DNN, the gradient of the log-policy w.r.t. the parameters can be obtained using backpropagation [38]. For VQCs executed on quantum hardware one typically resorts to the parameter-shift rule [39, 40].

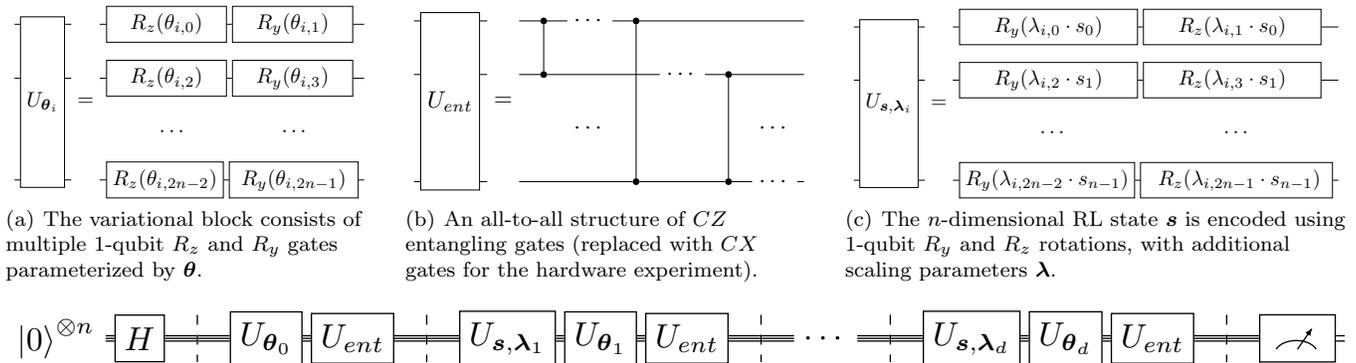


FIG. 2. Hardware-efficient quantum circuit, adapted from Jerbi et al. [21]. The parameters are summarized in  $\Theta = (\theta, \lambda)$ . Depending on circuit depth  $d$ , the encoding blocks  $U_{s, \lambda}$ , combined with variational blocks  $U_{\theta}$  and entanglement blocks  $U_{ent}$ , are repeated (i.e. for  $d \geq 2$  data re-uploading [28] is used). Measurements are performed in the computational basis.

### A. VQC-Model Architecture

We use a VQC with a subsequent measurement as a replacement for the DNN, which usually approximates the policy in deep RL [21]. The VQC acts on  $|0\rangle^{\otimes n}$  (where  $|0\rangle$  denotes the 1-qubit computational zero state) with the unitary  $U_{s, \lambda, \theta}$ , which prepares the quantum state  $|\psi_{s, \lambda, \theta}\rangle$ . We introduce the prerequisites of QC required for understanding our investigations and refer the interested reader to Nielsen and Chuang [33] for more details.

For the ansatz for our gate sequence, we closely follow Jerbi et al. [21] and use the hardware-efficient ansatz shown in Fig. 2. Besides the variational parameters  $\theta$ , there are also trainable scaling parameters  $\lambda$ , which enhance the expressivity of the model. For ease of notation we summarize both parameter sets as  $\Theta = (\theta, \lambda)$ .

Extracting classical information from the quantum state  $|\psi_{s, \Theta}\rangle$  is crucial when using VQCs in a hybrid algorithm. Usually, we measure some Hermitian operator  $O$  to estimate the expectation  $\langle \psi_{s, \Theta} | O | \psi_{s, \Theta} \rangle = \langle O \rangle_{s, \Theta}$ . For a projective measurement and spectral decomposition  $O = \sum_i \lambda_i |v_i\rangle \langle v_i|$ , the post-measurement state corresponds to one of the eigenstates  $|v_i\rangle$ , and we observe the corresponding eigenvalue  $\lambda_i$  with probability  $\langle \psi_{s, \Theta} | v_i \rangle \langle v_i | \psi_{s, \Theta} \rangle$ . We restrict our considerations to measuring only 1-qubit Pauli observables  $\sigma \in \{X, Y, Z\}$ .

### B. Reformulation of the RAW-VQC Policy

There are proposals of different measurement techniques, defining the RAW-VQC and the SOFTMAX-VQC policy [21]. While Jerbi et al. [21] suggest, that the latter formulation is superior in terms of RL performance, we argue in App. A, that it has serious drawbacks w.r.t. circuit sampling complexity. We experimentally demonstrate in Sec. V, that the RL performance of an improved version of the RAW-VQC policy is competitive. As the original definition is impractical for the upcoming discussions, we introduce a slight reformulation:

**Definition 1 (RAW-VQC).** *Given a VQC acting on  $n$  qubits, taking as input an RL state  $s \in \mathbb{R}^n$  and rotation angles  $\Theta \in [-\pi, \pi]^{|\Theta|}$ , such that it produces the quantum state  $|\psi_{s, \Theta}\rangle = U_{s, \Theta} |0\rangle^{\otimes n}$ , we define its associated RAW-VQC policy as:*

$$\pi_{\Theta}(a | s) = \langle P_a \rangle_{s, \Theta}, \quad (2)$$

where  $\langle P_a \rangle_{s, \Theta} = \langle \psi_{s, \Theta} | P_a | \psi_{s, \Theta} \rangle$  is the expectation value of a projector  $P_a$ . It must hold  $P_a = \sum_{|v\rangle \in \mathcal{V}_a} |v\rangle \langle v|$ , with  $\mathcal{V}_a \subseteq \mathcal{V}$ , where  $\mathcal{V} = \{|v_0\rangle, |v_1\rangle, \dots, |v_{2^n-1}\rangle\}$  is the set of eigenstates of an observable

$$O = \sum_{i=0}^{2^n-1} i \cdot |v_i\rangle \langle v_i|. \quad (3)$$

Furthermore, it must hold  $\bigcup_{a \in \mathcal{A}} \mathcal{V}_a = \mathcal{V}$ , and  $\mathcal{V}_i \cap \mathcal{V}_j = \emptyset$  for all  $i \neq j$ .

To be concise, the given reformulation is a bit more restrictive than the original one, due to the explicit designation of the eigenvalues in Eq. (3). This ensures, that by measuring eigenvalue  $i$ , we can directly conclude the post-measurement state to be  $|v_i\rangle$ . However, Def. 1 is completely equivalent in terms of all considerations and experiments carried out in Jerbi et al. [21].

## IV. ANALYSIS OF ACTION DECODING

We now focus on the action decoding scheme of the RAW-VQC policy. Def. 1 is instantiated such that the observable in Eq. (3) can be efficiently replaced with only 1-qubit Pauli operators. We start by measuring a 1-qubit Pauli observable  $\sigma_{n-1}$  on the uppermost qubit. The observed result is one of the eigenvalues  $\lambda_0 = +1$  or  $\lambda_1 = -1$ , which is interpreted as the bit value  $b_{n-1} = \frac{1-\lambda}{2}$ . The measured qubit is in the corresponding post-measurement state, while the other qubits have not been touched thus far. Now, we measure the Pauli observable

$\sigma_{n-2}$  on the next to uppermost qubit, and proceed this way until we have measured all the qubits. As all the 1-qubit Pauli observables on different qubits commute, the successive projections can be collected into one overall projection onto the respective basis state. The combined measurement result is the bitstring  $b_{n-1}b_{n-2}\cdots b_0$ , which is the binary expansion of  $i$  in Eq. (3). We follow the convention that the most significant bit corresponds to the uppermost wire of Fig. 2.

### A. Partitioning of Computational Basis States

Since measurements can typically only be done in the energy eigenbasis of a qubit, we select the Pauli operators  $\sigma_i$  to be all Pauli-Z observables. This boils down to a measurement in the *computational basis*, which for an  $n$ -qubit system is given by  $\mathcal{V} = \{|0\cdots 00\rangle, |0\cdots 01\rangle, \dots, |1\cdots 11\rangle\}$ . However, our techniques can also be applied to more general combinations of Pauli operators, as all share eigenvalues  $+1$  and  $-1$ . As instructed by Def. 1, all elements of this set have to be distributed among the available actions, i.e.,  $\mathcal{V}_a = \{|a_0\rangle, |a_1\rangle, \dots\} \subseteq \mathcal{V}$  for action  $a$ . Let the prepared state be represented in the computational basis (using decimal notation) as  $|\psi_{\mathbf{s},\Theta}\rangle = c_0|0\rangle + c_1|1\rangle + \dots + c_{N-1}|N-1\rangle$ , with  $N = 2^n$ , and  $\sum_i |c_i|^2 = 1$ . This allows the following reformulation of Eq. (2) in terms of the absolute squared amplitudes of the prepared quantum state:

$$\pi_{\Theta}(a|\mathbf{s}) = \left\langle \psi_{\mathbf{s},\Theta} \left| \sum_{|v\rangle \in \mathcal{V}_a} |v\rangle \langle v| \right| \psi_{\mathbf{s},\Theta} \right\rangle \quad (4)$$

$$= \sum_{|v\rangle \in \mathcal{V}_a} |c_v|^2. \quad (5)$$

Consequently, (as only Pauli observables are considered) it is sufficient to just sum up the absolute squared amplitudes associated with the respective basis states following Eq. (5) to determine the policy. This has to be done a bit differently on quantum hardware, as one can not directly access the amplitudes of a quantum state. However, it is possible to estimate the absolute squared value by executing the experiment multiple times. It corresponds to the probability of observing the eigenvalues associated with the respective basis states. With Pauli-Z observables, measuring an eigenvalue of  $i$  (which happens with probability  $|c_i|^2$ ) indicates the post-measurement state to be  $|i\rangle$ . In practice the measurement result is the binary expansion of  $i$ , i.e., the bitstring  $b_{n-1}b_{n-2}\cdots b_0$ .

For interacting with the environment, the RL agent needs to select an action according to the current policy  $\pi_{\Theta}(a|\mathbf{s})$ . Starting from Eq. (5), it holds  $\sum_{a \in \mathcal{A}} \sum_{|v\rangle \in \mathcal{V}_a} |c_v|^2 = 1$ . As all individual summands are non-negative, this defines a probability density function. Hence, it is sufficient to only measure the quantum state once. The agent decides for an action, based on

which partition  $\mathcal{V}_a$  the post-measurement state is contained within. For the parameter update, we must obtain  $\nabla_{\Theta} \ln \pi_{\Theta}(a|\mathbf{s}) = \nabla_{\Theta} \langle P_a \rangle_{\mathbf{s},\Theta} / \langle P_a \rangle_{\mathbf{s},\Theta}$ , for a trajectory of concrete instances of  $\mathbf{s}$  and  $a$ . To estimate  $\langle P_a \rangle_{\mathbf{s},\Theta}$  (and also  $\nabla_{\Theta} \langle P_a \rangle_{\mathbf{s},\Theta}$ ), we need to determine for each post-measurement state  $|i\rangle$ , if it is an element of  $\mathcal{V}_a$ .

There is a caveat with the explicit representation of basis state partitionings [21]. For larger systems, storing all  $\mathcal{V}_a$  is not feasible, as the total number of elements scales exponentially in  $n$ , independently of whether one considers quantum or classical computers (already for  $n = 64$  qubits, there are  $2^{64}$  possible bit strings, which would require  $2^{64} \cdot \log_2(64)$  bit  $\approx 147.6$  exabyte of storage space). A solution would be to use a classically computable post-processing function based on the measurement outputs. One expects that the RL performance may strongly depend on the choice of post-processing function. In the sequel we introduce a measure of globality for post-processing functions and provide strong evidence that it is directly correlated with RL performance.

### B. Action Decoding with Classical Post-Processing Function

To discuss the influence of post-processing functions on the RL performance, we start with the measured bitstrings  $b_{n-1}b_{n-2}\cdots b_1b_0$ . We denote the set of all bitstrings of length  $n$  by  $\mathcal{C}$ , and partition it into disjoint, action-associated sets  $\mathcal{C}_a$ . We define a *classical post-processing function*  $f_{\mathcal{C}} : \{0,1\}^n \rightarrow \{0,1,\dots,|\mathcal{A}|-1\}$ , such that  $f_{\mathcal{C}}(\mathbf{b}) = a$ , iff  $\mathbf{b} \in \mathcal{C}_a$  for an underlying partitioning of  $\mathcal{C}$ . This allows to reformulate Eq. (5) as

$$\pi_{\Theta}(a|\mathbf{s}) = \sum_{\mathbf{b} \in \{0,1\}^n} \langle \psi_{\mathbf{s},\Theta} | \mathbf{b} \rangle \langle \mathbf{b} | \psi_{\mathbf{s},\Theta} \rangle \quad (6)$$

$$\approx \frac{1}{K} \cdot \sum_{k=0}^{K-1} \delta_{f_{\mathcal{C}}(\mathbf{b}^{(k)})=a} \quad (7)$$

where  $K \gg 1$  is the number of shots for estimating the expectation value (Eq. (7) becomes exact for  $K \rightarrow \infty$ ),  $\mathbf{b}^{(k)}$  is the bitstring observed in the  $k$ -th shot, and  $\delta$  is an indicator function.

#### 1. Extracted Information Defines Globality Measure

In order to derive a quality measure for a specific post-processing function  $f_{\mathcal{C}}$ , we first define the notion of *extracted information* for an observed bitstring  $\mathbf{b}$ :

**Definition 2** (extracted information). *Let  $f_{\mathcal{C}}$  be a classical post-processing function, with a partitioning of the set of  $n$ -bit strings  $\mathcal{C} = \bigcup_a \mathcal{C}_a$ , for which  $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$  for all  $i \neq j$ . Furthermore,  $\mathbf{b} = b_{n-1}b_{n-2}\cdots b_1b_0$  denotes an arbitrary bitstring. The extracted information  $EI_{f_{\mathcal{C}}}(\mathbf{b}) \in \mathbb{N}$*

is the minimum number of bits  $b_i$  necessary, to compute  $f_C(\mathbf{b})$ , i.e. assign  $\mathbf{b}$  unambiguously to a set  $\mathcal{C}_a$ .

An example of a valid partitioning associated with a 4-qubit system and  $|\mathcal{A}| = 4$  is

$$\mathcal{C}_{a=0} = \{0000, 0010, 0100, 0110\}, \quad (8)$$

$$\mathcal{C}_{a=1} = \{0001, 0011, 0101, 0111\}, \quad (9)$$

$$\mathcal{C}_{a=2} = \{1000, 1010, 1101, 1111\}, \quad (10)$$

$$\mathcal{C}_{a=3} = \{1001, 1011, 1100, 1110\}. \quad (11)$$

For assigning the bitstring 0111 unambiguously to the correct set, i.e.  $\mathcal{C}_{a=1}$ , it is enough to consider only the first and the last bit. It is straightforward to see that it cannot work with less information. As all partitions contain the same number of elements, and we need to choose between 4 actions, it requires at least  $\log_2(4) = 2$  bits of information. Therefore, the extracted information is given by  $EI_{f_C}(0111) = 2$ .

To grant more expressivity to the defined measure, we average the extracted information over all possible bitstrings to get the *globality measure*

$$G_{f_C} := \frac{1}{2^n} \sum_{\mathbf{b} \in \{0,1\}^n} EI_{f_C}(\mathbf{b}). \quad (12)$$

The value of this measure describes the average amount of information (in bits) necessary to have an unambiguous distinction between the different actions. The concept is inspired by the reformulation of special policies using local and global observables in App. B.

A lower bound to  $G_{f_C}$  is intuitively given by  $G_{f_C} \geq \log_2(|\mathcal{A}|)$ . The measure is trivially upper bounded by  $G_{f_C} \leq n$ , which is in line with Holevo's theorem [33, 41], in that no more than  $n$  bits of classical information can be extracted from a  $n$ -qubit system in a single measurement. Evaluating Eq. (12) for the example above gives  $G_{f_C} = 2.5$ , i.e., on average 2.5 bits of information are necessary (see App. C for the exact computation).

Evaluating Eq. (12) requires simulating the quantum system, which is infeasible for large  $n$ . Also with access to such a routine, determining the extracted information for a concrete bitstring poses a non-trivial task. Nonetheless, one can define a post-processing function, which has maximal globality according to Eq. (12). We discuss this construction in the next section.

## 2. Constructing an Optimal Post-Processing Function

As we demonstrate in Sec. V, the value of the introduced globality measure is strongly correlated with the RL performance. It is not feasible to construct a post-processing function with optimal globality measure using a brute-force approach, as we argue in App. D 1. To circumvent this caveat, we construct an implicit partitioning  $\mathcal{C}$ , that gives rise to a post-processing function with provably optimal globality  $G_{f_C} = n$ . The set of bitstrings

$\mathbf{b} = b_{n-1} \cdots b_0$  associated with action  $a$  is recursively defined as

$$\mathcal{C}_{[a]_2}^{(m)} = \left\{ \mathbf{b} \mid \bigoplus_{i=m}^{n-1} b_i = a_0 \wedge \mathbf{b} \in \mathcal{C}_{a_m \cdots a_2(a_1 \oplus a_0)}^{(m-1)} \right\} \quad (13)$$

where  $m = \log_2(M) - 1$  (with  $M := |\mathcal{A}|$ ) and  $[a]_2 = a_m \cdots a_0$  is the binary expansion of  $a$ . The base cases use a binary parity function on all bits:

$$\mathcal{C}_{[0]_2}^{(0)} = \left\{ \mathbf{b} \mid \bigoplus_{i=0}^{n-1} b_i = 0 \wedge \mathbf{b} \in \{0,1\}^n \right\} \quad (14)$$

$$\mathcal{C}_{[1]_2}^{(0)} = \left\{ \mathbf{b} \mid \bigoplus_{i=0}^{n-1} b_i = 1 \wedge \mathbf{b} \in \{0,1\}^n \right\} \quad (15)$$

The construction in Eq. (13) thus recursively splits Eqs. (14) and (15) by computing parity values of sub-strings, until the required number of groups is formed.

**Lemma 1.** *Let an arbitrary VQC act on an  $n$ -qubit state. The RAW-VQC policy needs to distinguish between  $M := |\mathcal{A}|$  actions, where  $M$  is a power of 2, i.e.,  $m = \log_2(M) - 1 \in \mathbb{N}_0$ . Using Eqs. (13) to (15) we define*

$$\pi_{\Theta}^{glob}(a | \mathbf{s}) = \sum_{v \in \mathcal{C}_{[a]_2}^{(m)}} \langle \psi_{\mathbf{s}, \Theta} | v \rangle \langle v | \psi_{\mathbf{s}, \Theta} \rangle \quad (16)$$

$$\approx \frac{1}{K} \sum_{k=0}^{K-1} \delta_{f_C^{(m)}(\mathbf{b}^{(k)})=a} \quad (17)$$

where  $K$  is the number of shots for estimating the expectation value,  $\mathbf{b}^{(k)}$  is the bitstring observed in the  $k$ -th shot, and  $\delta$  is an indicator function. The post-processing function is guaranteed to have the globality value  $G_{f_C} = n$ .

The proof is deferred to App. D. This post-processing function defines the proposed QPG algorithm in Fig. 1.

By construction of the recursive definition the action associated with a specific bitstring  $f_C(\mathbf{b})$  is given as

$$f_C(\mathbf{b}) = \left[ b_0 \cdots b_{m-1} \left( \bigoplus_{i=m}^{n-1} b_i \right) \right]_{10} \quad (18)$$

where  $[\cdot]_{10}$  denotes the decimal representation of the respective bitstring. This representation does not require storing the partitioning  $\mathcal{C}$  explicitly, which renders our approach feasible for large system sizes and action spaces.

## V. EXPERIMENTS

Our framework realizes different post-processing functions, associated with various globality measure values. The implementation is based upon the `qiskit` and `qiskit_machine_learning` libraries. If not stated differently, all experiments use the `StatevectorSimulator`,

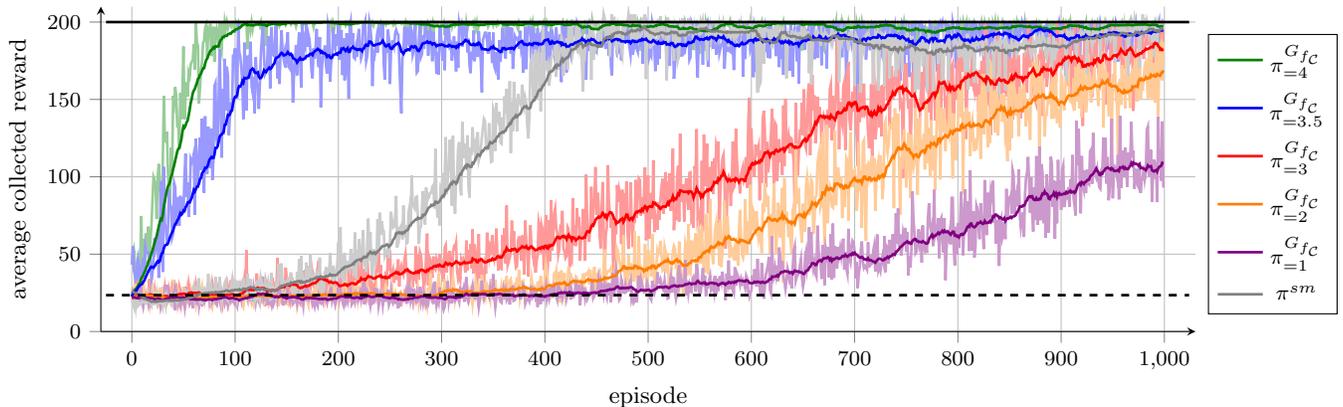


FIG. 3. RL training performance of RAW-VQC policies with different post-processing functions on the `CartPole-v0` environment, averaged over 10 independent runs and 20 preceding timesteps (dark curves). A higher globality value correlates with a faster convergence to the optimal collected reward of 200. For comparison, the performance of a `SOFTMAX-VQC` is included as  $\pi^{sm}$ .

which assumes the absence of noise, and also eliminates sampling errors. The computations were executed on a CPU-cluster with 64 nodes, each equipped with 4 cores and 32 GB of working memory. The experiments in Subsecs. VA and VB focus on the `CartPole` environment, while Subsec. VC and App. F also consider `ContextualBandits` and `FrozenLake`, respectively. We establish conventions regarding the experimental setup and reproducibility in App. F.

### A. RL Performance vs. Globality Measure

The main experiment is conducted on the `CartPole-v0` environment [42] with a horizon of 200 time-steps. The state space has a dimensionality of 4, with all values scaled to be within  $[-1, 1)$ . The agent can take two actions, i.e.,  $|\mathcal{A}| = 2$ .

All experiments in Fig. 3 (apart from the `SOFTMAX-VQC` policy  $\pi^{sm}$ , which uses a tensored Pauli-Z measurement on all qubits [21]) use the same architecture, only the post-processing function is modified. The two extreme cases are  $G_{fc} = 4.0$ , constructed following Lem. 1, and  $G_{fc} = 1.0$ , which extracts the lowest amount of information that is sufficient. We also experiment with  $G_{fc} = 2.0$  and  $G_{fc} = 3.0$ , both of which can be expressed as a parity measurement on 2 or 3 qubits, respectively. A special case is  $G_{fc} = 3.5$ , where the explicit partitioning  $\mathcal{C}_{a=0}^{3.5} = \{1, 3, 5, 6, 9, 10, 12, 15\}$  and  $\mathcal{C}_{a=1}^{3.5} = \{0, 2, 4, 7, 8, 11, 13, 14\}$  is used.

Throughout all considerations, the RL performance clearly benefits from a higher globality value of the underlying post-processing function. The convergence speed is improved, for example, the strategy learned by an agent with an underlying global post-processing function reaches optimal behavior after just 100 episodes. This is clearly delayed for all other configurations. In fact, the policy with  $G_{fc} = 1.0$  is not able to learn optimal behavior, even after 5,000 episodes. This can be partially addressed with deeper circuits (see App. F 1). However,

as circuit depth is very critical for NISQ devices, using optimal classical post-processing functions is crucial.

Fig. 3 also depicts the performance of a `SOFTMAX-VQC` policy [21]. Interestingly, it performs better than RAW-VQC with a globality value  $\leq 3$ , but is clearly inferior to the two fastest converging setups.

It is important to mention, that this overall behavior cannot only be observed in this concrete setup and environment. We obtained comparable results on `CartPole-v1`, which extends the horizon to 500 steps. Results on further environments, namely `FrozenLake` and `ContextualBandits`, are provided in App. F 2.

### B. Analysis of Effective Dimension and Fisher Information Spectrum

For every machine learning task, two crucial factors are the *expressibility* and *trainability* of the used model. Tools for quantitative analysis, based on the Fisher information matrix (FIM), have recently been proposed by Abbas et al. [43], and a comparative study of various quantum neural network architectures has been conducted in Wilkinson et al. [44]. An adaption of those concepts to the RL setup is deferred to App. E.

The expressibility of a model can be quantified using the *effective dimension* [43], which describes the variety of functions that can be approximated. A normalized version of the effective dimension for different RAW-VQC policy setups is compared in Fig. 4(a). The expressive power of the respective model is proportional to the globality of the underlying post-processing function and also the RL performance from Fig. 3. The policies with  $G_{fc} = 4$  and  $G_{fc} = 3.5$  pose an exception, as the respective effective dimensions coincide. We considered this statistical variance, as also the RL performance varies only slightly. While a more expressive circuit provides no guarantee of better performance, a complex problem needs a model that is expressive enough, promoting the usage of post-processing functions with high globality.

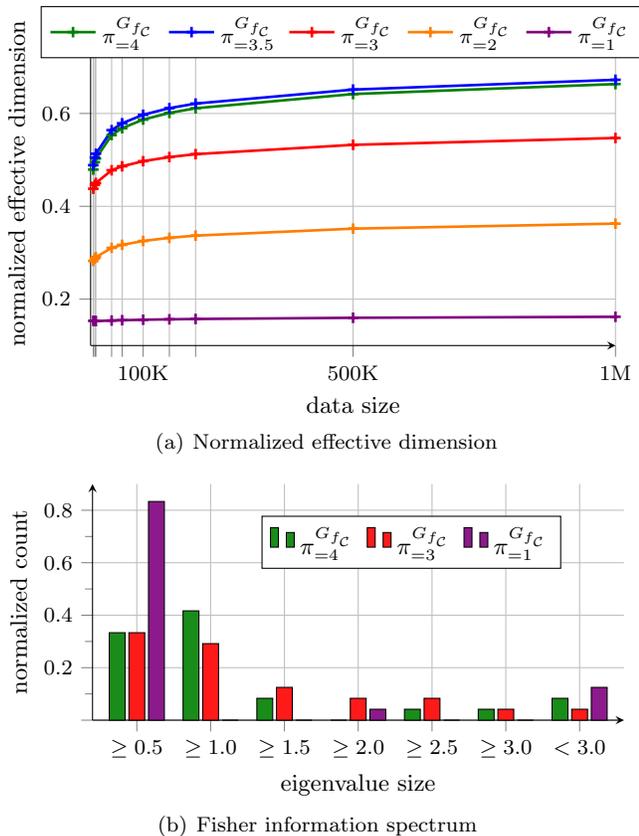


FIG. 4. Quantities related to expressibility and trainability of a VQC-based model. We estimate the FIM with 100 random parameter sets for each of the 100 random states  $\mathbf{s}$ . We draw the elements of each state from  $\mathcal{N}(0, 0.5)$ , which mimics the prior state distribution of the `CartPole` environment.

Insights into the trainability of a model are possible by considering the spectrum of the FIM [43], which captures the geometry of the parameter space. Trainability profits from a uniform spectrum, while distorted spectra are suboptimal. Fig. 4(b) depicts the Fisher information spectrum for post-processing functions with  $G_{fc} = 4.0$ ,  $G_{fc} = 3.0$ , and  $G_{fc} = 1.0$ . There is no clear difference between the Fisher information spectra associated with the ones with higher globality. The difference to the least-global configuration is more significant, where most of the eigenvalues are close to 0. This implies that the parameter space is flat in most dimensions, making optimization difficult. Additionally, there are a few large eigenvalues, indicating a distorted optimization space. In absolute terms, the spectra for  $\pi_{=4}^{G_{fc}}$  and  $\pi_{=3}^{G_{fc}}$  are not uniform. However, in comparison to  $\pi_{=1}^{G_{fc}}$ , the eigenvalues are much more uniformly distributed, and also fewer outliers exist. This property becomes more significant when considering larger system sizes and circuit depths, see App. G. Hence, the globality value associated with a model correlates at least to some extent with the uniformity of the Fisher information spectrum, which is beneficial for trainability.

### C. Training on Quantum Hardware

To emphasize the practical relevance of our method, we conducted an experiment on actual quantum hardware. There is work on VQC-based RL that performs the training on classical hardware and then uploads the learned parameters to quantum hardware for testing [24, 45], where the trained models can replicate the learned behavior on the hardware to some extent. We take a more involved approach and perform both training and testing on quantum hardware. To the best of our knowledge, this is the first investigation of VQC-based RL on quantum hardware that also includes the training routine.

We select an 8-state `ContextualBandits` environment [36] for the experiment, which can be implemented with a 3-qubit system. The employed hardware backend is the 5-qubit device `ibmq_manila` [46]. We slightly adapted the VQC architecture and typical RL feedback loop to make hardware usage feasible. First, we replaced the `CZ` gates from Fig. 2 with `CX` gates, due to the former one not being hardware-native (which would lead to decomposition and additional circuit complexity). Second, to reduce the number of hardware uploads, we used a batch size of 50 trajectories. Consequently the gradients, and therefore also the parameter updates, are only computed for each 50th time-step. Still, for a horizon of 1,000 episodes, this adds up to 180 individual uploads and overall 9,760 expectation values that need to be estimated. With 1,024 shots to estimate each one, close to 10M circuits had to be evaluated. This took approximately 30 hours to complete. We have to note at this point, that the actual computation time accounted only for a small fraction, as the main bottleneck was the queuing system and the waiting time between uploads.

The training performance for this setup is displayed in Fig. 5. We compare it to the results obtained from simulation on classical hardware, without and with simulated noise. The `qiskit` noise model was instantiated with parameters sampled from the `ibmq_manila` device. The noise-free simulation clearly produces the best results and is able to learn an optimal policy. While this is not the case for the experiments with simulated noise and on hardware, there is still a clear improvement over the initial random policy. However, not much progress is made after approximately 400 episodes. The testing results in Tab. I clearly show that the hardware-trained and tested agent is able to identify the optimal action for all 8 states. The problem seems to be that the policy does not get “peaky” enough. We assume this is due to readout errors  $\varepsilon_M$  and noise mainly induced by entangling gates  $\varepsilon_{CX}$ . While the original circuit uses only 15 `CX` gates, the transpiled versions average to about 27, due to the sparse connectivity structure of the hardware device. The noise can potentially be countered by using an architecture more adapted to the basis gate set and connectivity structure. Apart from that, error mitigation strategies [47] could be a suitable option. It has to be noted that the overall length of the transpiled circuit

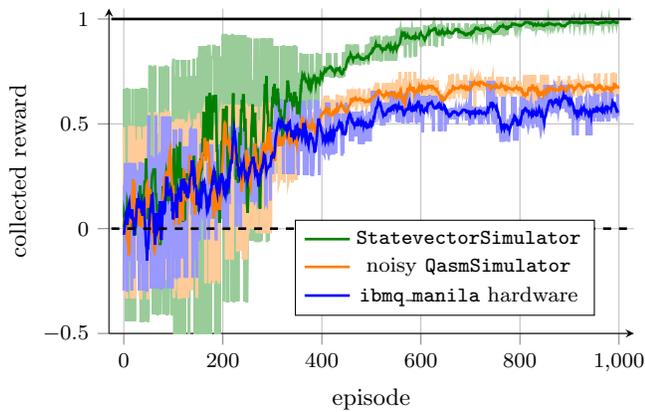


FIG. 5. Training performance of a RAW-VQC policy with maximum globality value on an 8-state and 2-action `ContextualBandits` environment. Two training runs are executed on classical hardware (wo/ and w/ simulated noise).

	$s = 0$	$s = 1$	$s = 2$	$s = 3$
optimal action	$a = 0$	$a = 0$	$a = 1$	$a = 1$
$\pi(a = 0   s)$	<b>0.83</b>	<b>0.85</b>	0.17	0.18
$\pi(a = 1   s)$	0.17	0.15	<b>0.83</b>	<b>0.82</b>
	$s = 4$	$s = 5$	$s = 6$	$s = 7$
optimal action	$a = 0$	$a = 0$	$a = 1$	$a = 1$
$\pi(a = 0   s)$	<b>0.83</b>	<b>0.72</b>	0.22	0.20
$\pi(a = 1   s)$	0.17	0.28	<b>0.78</b>	<b>0.80</b>

TABLE I. RL testing results on the `ibmq_manila` hardware device (with model trained on the same hardware in Fig. 5). In all cases, the agent is able to identify the optimal action. As for the `ContextualBandits` environment the states are sampled uniformly at random in every step, the optimal action is selected in approx. 81% of all cases, which is a clear improvement over using a random policy.

stays approximately constant throughout all episodes, consequently, it does not explain the performance gap in the later stages. Also the calibration of the system after episode 200 (increasing  $\varepsilon_M$  from 2.98% to 4.61% and  $\varepsilon_{CX}$  from 0.74% to 0.93%) did not cause a clear change in performance. The noisy simulation approximates the hardware results quite well, although there is still a small gap. This might be down to the reason, that the error model underestimates the crosstalk errors, which slightly improves the performance.

## VI. CONCLUSION

This paper analyzed the action decoding procedure of the quantum policy gradient (QPG) algorithm originally proposed by Jerbi et al. [21]. We proposed a hybrid routine combining measurements in the computational basis and a classical post-processing function. A newly developed globality measure for those functions showed a

strong correlation with the reinforcement learning (RL) performance and some model complexity measures. Most importantly, we provided a routine to implement a post-processing function that is optimal with respect to this measure. The function is also feasible for large action spaces. Compared to the original RAW-variational quantum circuit (VQC), as well as the `SOFTMAX-VQC` policy [21], we achieve significant gains in RL performance, with only negligible classical overhead. With this enhanced QPG algorithm, we are able to successfully execute the entire RL training and testing routine on actual quantum hardware.

Our work focused on RL routines, but we expect that our findings can be extended to the realm of supervised and unsupervised learning. While we did not claim any experimental quantum advantage in our paper, the provable advantage for an environment constructed based on the discrete logarithm from Jerbi et al. [21] also holds for our approach.

## ACKNOWLEDGEMENTS

We wish to thank G. Wellein for his administrative and technical support of this work. The authors gratefully acknowledge the scientific support and HPC resources provided by the Erlangen National High Performance Computing Center (NHR@FAU) of the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU). The hardware is funded by the German Research Foundation (DFG).

We acknowledge the use of IBM Quantum services for this work. The views expressed are those of the authors, and do not reflect the official policy or position of IBM or the IBM Quantum team.

**Funding.** The research was supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy with funds from the Hightech Agenda Bayern and by the Bavarian Ministry for Economic Affairs, Infrastructure, Transport and Technology through the Center for Analytics-Data-Applications (ADA Center) within the framework of “BAYERN DIGITAL II”.

M. Hartmann acknowledges support by the European Union’s Horizon 2020 research and innovation programme under grant agreement No 828826 “Quomorphic” and the Munich Quantum Valley, which is supported by the Bavarian state government with funds from the Hightech Agenda Bayern Plus.

## CODE AVAILABILITY

A repository with the framework to reproduce the main results of this paper is available at [https://gitlab.cc-asp.fraunhofer.de/meyerno/qpg\\_algorithm\\_with\\_optimized\\_action\\_decoding](https://gitlab.cc-asp.fraunhofer.de/meyerno/qpg_algorithm_with_optimized_action_decoding). Further information and data is available upon reasonable request.

- 
- [1] S. M. Kakade, *On the Sample Complexity of Reinforcement Learning*, Ph.D. thesis, University College London (2003).
- [2] M. A. Nielsen, *Neural Networks and Deep Learning* (Determination Press, 2015).
- [3] T. Poggio, A. Banburski, and Q. Liao, Theoretical issues in deep networks, *Proceedings of the National Academy of Sciences* **117**, 30039 (2020).
- [4] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, and J. Zhang, End to end learning for self-driving cars, arXiv:1604.07316 10.48550/arXiv.1604.07316 (2016).
- [5] T. Maksymyuk, J. Gazda, O. Yaremko, and D. Nevinskiy, Deep learning based massive MIMO beamforming for 5G mobile network, in *International Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS)* (2018) pp. 241–244.
- [6] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, *et al.*, Quantum supremacy using a programmable superconducting processor, *Nature* **574**, 505 (2019).
- [7] H.-S. Zhong, H. Wang, Y.-H. Deng, M.-C. Chen, L.-C. Peng, Y.-H. Luo, J. Qin, D. Wu, X. Ding, Y. Hu, *et al.*, Quantum computational advantage using photons, *Science* **370**, 1460 (2020).
- [8] Y. Wu, W.-S. Bao, S. Cao, F. Chen, M.-C. Chen, X. Chen, T.-H. Chung, H. Deng, Y. Du, D. Fan, *et al.*, Strong quantum computational advantage using a superconducting quantum processor, *Phys. Rev. Lett.* **127**, 180501 (2021).
- [9] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, Parameterized quantum circuits as machine learning models, *Quantum Sci. Technol.* **4**, 043001 (2019).
- [10] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, *et al.*, Variational quantum algorithms, *Nat. Rev. Phys.* **3**, 625 (2021).
- [11] K. Bharti, A. Cervera-Liarta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, *et al.*, Noisy intermediate-scale quantum algorithms, *Rev. Mod. Phys.* **94**, 015004 (2022).
- [12] J. Li, X. Yang, X. Peng, and C.-P. Sun, Hybrid Quantum-Classical Approach to Quantum Optimal Control, *Phys. Rev. Lett.* **118**, 150503 (2017).
- [13] N. Moll, P. Barkoutsos, L. S. Bishop, J. M. Chow, A. Cross, D. J. Egger, S. Filipp, A. Fuhrer, J. M. Gambetta, M. Ganzhorn, *et al.*, Quantum optimization using variational algorithms on near-term quantum devices, *Quantum Sci. Technol.* **3**, 030503 (2018).
- [14] K. Sharma, S. Khatry, M. Cerezo, and P. J. Coles, Noise resilience of variational quantum compiling, *New J. Phys.* **22**, 043006 (2020).
- [15] E. Fontana, N. Fitzpatrick, D. M. Ramo, R. Duncan, and I. Rungger, Evaluating the noise resilience of variational quantum algorithms, *Phys. Rev. A* **104**, 022403 (2021).
- [16] T. Goto, Q. H. Tran, and K. Nakajima, Universal Approximation Property of Quantum Machine Learning Models in Quantum-Enhanced Feature Spaces, *Phys. Rev. Lett.* **127**, 090506 (2021).
- [17] M. Schuld, R. Sweke, and J. J. Meyer, Effect of data encoding on the expressive power of variational quantum-machine-learning models, *Phys. Rev. A* **103**, 032430 (2021).
- [18] Y. Liu, S. Arunachalam, and K. Temme, A rigorous and robust quantum speed-up in supervised machine learning, *Nat. Phys.* **17**, 1013 (2021).
- [19] R. Sweke, J.-P. Seifert, D. Hangleiter, and J. Eisert, On the Quantum versus Classical Learnability of Discrete Distributions, *Quantum* **5**, 417 (2021).
- [20] H.-Y. Huang, R. Kueng, and J. Preskill, Predicting many properties of a quantum system from very few measurements, *Nat. Phys.* **16**, 1050 (2020).
- [21] S. Jerbi, C. Gyurik, S. Marshall, H. Briegel, and V. Dunjko, Parametrized Quantum Policies for Reinforcement Learning, *Adv. Neural Inf. Process. Syst.* **34**, 28362 (2021).
- [22] N. Meyer, C. Ufrecht, M. Periyasamy, D. D. Scherer, A. Plinge, and C. Mutschler, A Survey on Quantum Reinforcement Learning, arXiv:2211.03464 10.48550/arXiv.2211.03464 (2022).
- [23] V. Saggio, B. E. Asenbeck, A. Hamann, T. Strömberg, P. Schiansky, V. Dunjko, N. Friis, N. C. Harris, M. Hochberg, D. Englund, *et al.*, Experimental quantum speed-up in reinforcement learning agents, *Nature* **591**, 229 (2021).
- [24] S. Y.-C. Chen, C.-H. H. Yang, J. Qi, P.-Y. Chen, X. Ma, and H.-S. Goan, Variational Quantum Circuits for Deep Reinforcement Learning, *IEEE Access* **8**, 141007 (2020).
- [25] A. Sequeira, L. P. Santos, and L. S. Barbosa, Variational Quantum Policy Gradients with an Application to Quantum Control, arXiv:2203.10591 10.48550/arXiv.2203.10591 (2022).
- [26] S. Wu, S. Jin, D. Wen, and X. Wang, Quantum reinforcement learning in continuous action space, arXiv:2012.10711 10.48550/arXiv.2012.10711 (2020).
- [27] Y. Kwak, W. J. Yun, S. Jung, J.-K. Kim, and J. Kim, Introduction to Quantum Reinforcement Learning: Theory and PennyLane-based Implementation, in *International Conference on Information and Communication Technology Convergence (ICTC)* (2021) pp. 416–420.
- [28] A. Pérez-Salinas, A. Cervera-Liarta, E. Gil-Fuster, and J. I. Latorre, Data re-uploading for a universal quantum classifier, *Quantum* **4**, 226 (2020).
- [29] M. Periyasamy, N. Meyer, C. Ufrecht, D. D. Scherer, A. Plinge, and C. Mutschler, Incremental Data-Uploading for Full-Quantum Classification, in *IEEE International Conference on Quantum Computing and Engineering (QCE)* (2022) pp. 31–37.
- [30] S. Sim, P. D. Johnson, and A. Aspuru-Guzik, Expressibility and Entangling Capability of Parameterized Quantum Circuits for Hybrid Quantum-Classical Algorithms, *Adv. Quantum Technol.* **2**, 1900070 (2019).
- [31] 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**, 242 (2017).
- [32] V. B. Braginsky, V. B. Braginski, and F. Y. Khalili, *Quantum measurement* (Cambridge University Press, 1995).

- [33] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, 2010).
- [34] M. Schuld and F. Petruccione, *Supervised Learning with Quantum Computers* (Springer, 2018).
- [35] M. Schuld, Supervised quantum machine learning models are kernel methods, arXiv:2101.11020 10.48550/arXiv.2101.11020 (2021).
- [36] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction* (MIT press, 2018).
- [37] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, Policy Gradient Methods for Reinforcement Learning with Function Approximation, in *Advances in Neural Information Processing Systems*, Vol. 12 (1999).
- [38] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Learning representations by back-propagating errors, *Nature* **323**, 533 (1986).
- [39] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, Quantum circuit learning, *Phys. Rev. A* **98**, 032309 (2018).
- [40] M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, and N. Killoran, Evaluating analytic gradients on quantum hardware, *Phys. Rev. A* **99**, 032331 (2019).
- [41] A. S. Holevo, Bounds for the Quantity of Information Transmitted by a Quantum Communication Channel, *Problems Inform. Transmission* **9**, 177 (1973).
- [42] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, OpenAI Gym, arXiv:1606.01540 10.48550/arXiv.1606.01540 (2016).
- [43] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, The power of quantum neural networks, *Nat. Comput. Sci.* **1**, 403 (2021).
- [44] S. A. Wilkinson and M. J. Hartmann, Evaluating the performance of sigmoid quantum perceptrons in quantum neural networks, arXiv:2208.06198 10.48550/arXiv.2208.06198 (2022).
- [45] J.-Y. Hsiao, Y. Du, W.-Y. Chiang, M.-H. Hsieh, and H.-S. Goan, Unentangled quantum reinforcement learning agents in the OpenAI Gym, arXiv:2203.14348 10.48550/arXiv.2203.14348 (2022).
- [46] IBM Quantum, <https://quantum-computing.ibm.com/> (2021).
- [47] S. Endo, Z. Cai, S. C. Benjamin, and X. Yuan, Hybrid Quantum-Classical Algorithms and Quantum Error Mitigation, *J. Phys. Soc. Japan* **90**, 032001 (2021).
- [48] M. Franz, L. Wolf, M. Periyasamy, C. Uftrecht, D. D. Scherer, A. Plinge, C. Mutschler, and W. Mauerer, Uncovering instabilities in variational-quantum deep Q-networks, *J. Franklin Inst.* 10.1016/j.jfranklin.2022.08.021 (2022).
- [49] D. P. Kingma and J. Ba, Adam: A Method for Stochastic Optimization, arXiv:1412.6980 10.48550/arXiv.1412.6980 (2014).
- [50] S. J. Reddi, S. Kale, and S. Kumar, On the Convergence of Adam and Beyond, arXiv:1904.09237 10.48550/arXiv.1904.09237 (2019).
- [51] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, Barren plateaus in quantum neural network training landscapes, *Nat. Commun.* **9**, 1 (2018).
- [52] S. Hochreiter, The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions, *Int. J. Uncertain. Fuzz.* **6**, 107 (1998).

## Appendix A: Caveats of the SOFTMAX-VQC Policy

In Subsec. III B we have stated, that using the SOFTMAX-VQC policy is problematic w.r.t. circuit sampling complexity. This was not explicitly addressed in Jerbi et al. [21], where experimental results suggest that the SOFTMAX-VQC policy formulation exhibits clearly superior performance in some simple benchmark environments, compared to the RAW-VQC policy. This is partially explained by the argument, that this formulation has better abilities in dealing with the balancing of exploration and exploitation. More concretely, this trade-off can be influenced by the inverse temperature parameter  $\beta$  in the SOFTMAX-VQC policy equation

$$\pi_{\lambda, \theta}(a | \mathbf{s}) = \frac{e^{\beta \langle O_a \rangle_{\mathbf{s}, \lambda, \theta}}}{\sum_{a' \in \mathcal{A}} e^{\beta \langle O_{a'} \rangle_{\mathbf{s}, \lambda, \theta}}}, \quad (\text{A.1})$$

where  $\langle O_a \rangle_{\mathbf{s}, \lambda, \theta} := \langle \psi_{\mathbf{s}, \lambda, \theta} | O_a | \psi_{\mathbf{s}, \lambda, \theta} \rangle$ , and  $O_a$  is some action-dependent observable.

There are two parts of the QPG pipeline, in which this formulation has an undesirable circuit sampling complexity. This is especially troublesome for the currently existing *nisq* devices, as every execution of a quantum circuit exhibits considerable costs. First of all, the action selection following Eq. (A.1) requires the estimation of  $|\mathcal{A}|$  different expectation values (unlike for the RAW-VQC, where a single measurement in the computational basis is sufficient). Secondly, also the approximation of the log-policy gradients scales linearly with the number of actions  $|\mathcal{A}|$ , as a different observable  $O_a$  is selected for each action  $a$  (unlike for the RAW-VQC, where the estimation of only one expectation value is sufficient). In order to avoid this dependence of the scaling on the size of the action space, most experiments in Jerbi et al. [21] use a fixed observable  $O$  for all actions, which gets multiplied with some action-dependent classical weight  $w_a$ . We formalize this approach in the following:

**Definition A.1 (RESTRICTED-SOFTMAX-VQC).** *Given a VQC acting on  $n$  qubits, taking as input a state  $\mathbf{s} \in \mathbb{R}^n$ , rotation angles  $\boldsymbol{\theta} \in [-\pi, \pi]^{|\boldsymbol{\theta}|}$ , and scaling parameters  $\boldsymbol{\lambda} \in \mathbb{R}^{|\boldsymbol{\lambda}|}$ , such that it produces the quantum state  $|\psi_{\mathbf{s}, \boldsymbol{\theta}, \boldsymbol{\lambda}}\rangle = U_{\mathbf{s}, \boldsymbol{\theta}, \boldsymbol{\lambda}} |0\rangle^{\otimes n}$ , we define its associated RESTRICTED-SOFTMAX-VQC policy as:*

$$\pi_{\Theta}(a | \mathbf{s}) = \frac{e^{\beta w_a \langle O \rangle_{\mathbf{s}, \lambda, \theta}}}{\sum_{a'} e^{\beta w_{a'} \langle O \rangle_{\mathbf{s}, \lambda, \theta}}} \quad (\text{A.2})$$

where  $\langle O \rangle_{\mathbf{s}, \lambda, \theta} = \langle \psi_{\mathbf{s}, \lambda, \theta} | O | \Psi_{\mathbf{s}, \lambda, \theta} \rangle$  is the expectation value of the observable  $O$ ,  $w_a$  is a weight parameter associated with action  $a$ , and  $\beta \in \mathbb{R}$  is an inverse-temperature parameter.  $\Theta = (\boldsymbol{\theta}, \boldsymbol{\lambda}, \mathbf{w})$  constitute all the trainable parameters of this policy.

For completeness, we demonstrate in App. A 1, that the circuit sampling complexity of this simplified version is only constant in the number of actions. A serious restriction of the RL performance caused by this simplification us derived in App. A 2. In App. A 3, we extract some implications for the original SOFTMAX-VQC policy from the findings.

### 1. Circuit Sampling Complexity of RESTRICTED-SOFTMAX-VQC Policy

It follows directly from Eq. (A.2), that the action selection procedure requires the estimation of only a single expectation value, namely  $\langle O \rangle_{\mathbf{s}, \lambda, \theta}$ . It is not directly obvious, if this reduction also translates to the gradient estimation, more concretely to the log-policy gradient required for Eq. (1). Following Def. A.1, the gradients w.r.t.  $\boldsymbol{\theta}$  of a RESTRICTED-SOFTMAX-VQC simplify to

$$\nabla_{\boldsymbol{\theta}} \ln \pi_{\Theta}(a | \mathbf{s}) \quad (\text{A.3})$$

$$= \beta \cdot \left( \nabla_{\boldsymbol{\theta}} w_a \langle O \rangle_{\mathbf{s}, \theta, \lambda} - \sum_{a'} \pi_{\Theta}(a' | \mathbf{s}) \cdot \nabla_{\boldsymbol{\theta}} w_{a'} \langle O \rangle_{\mathbf{s}, \theta, \lambda} \right) \quad (\text{A.4})$$

$$= \beta \cdot \nabla_{\boldsymbol{\theta}} \langle O \rangle_{\mathbf{s}, \theta, \lambda} \left( w_a - \sum_{a'} \pi_{\Theta}(a' | \mathbf{s}) \cdot w_{a'} \right). \quad (\text{A.5})$$

A similar derivation holds for  $\nabla_{\boldsymbol{\lambda}} \ln \pi_{\Theta}(a | \mathbf{s})$ . The gradient w.r.t. the weight associated with action  $x$  is given as

$$\nabla_{w_x} \ln \pi_{\Theta}(a | \mathbf{s}) \quad (\text{A.6})$$

$$= \beta \cdot \left( \nabla_{w_x} w_a \langle O \rangle_{\mathbf{s}, \theta, \lambda} - \sum_{a'} \pi_{\Theta}(a' | \mathbf{s}) \cdot \nabla_{w_x} w_{a'} \langle O \rangle_{\mathbf{s}, \theta, \lambda} \right) \quad (\text{A.7})$$

$$= \beta \cdot \langle O \rangle_{\mathbf{s}, \theta, \lambda} (\delta_{a=x} - \pi_{\Theta}(x | \mathbf{s})), \quad (\text{A.8})$$

with  $\delta_{a=x} = 1$ , iff  $a = x$ , and  $\delta_{a,x} = 0$  otherwise. Consequently, for both, action selection and gradient computation, only one observable has to be considered. This removes the dependence of the circuit sampling complexity on the number of actions, which must be avoided for environments with big action spaces.

## 2. Structural Restriction of RL Performance for RESTRICTED-SOFTMAX-VQC

While the previous considerations are quite promising when talking about circuit sampling complexity, there is also a serious drawback of the RESTRICTED-SOFTMAX-VQC approach. More concretely, it is not suitable for problems with big action spaces, as most information is only contained in the classical weight parameters. This statement is concertized and proven in the following. First of all, we restrict our initial considerations to a specific type of RL environment.

**Definition A.2** (uniform environment). *An environment  $\mathcal{E}_{\mathcal{A}}$  is considered uniform, iff it is solved by a deterministic policy, which is expected to select each distinct action the same amount of times. More explicitly, let  $\mathcal{S}_i$  denote a set of states from  $\mathcal{S}$ , with  $\bigcup_{\mathcal{A}} \mathcal{S}_i = \mathcal{S}$  and  $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset$  for all  $i \neq j$ . Following the optimal policy  $\pi_*$ , each of the state sets must be equally likely to observe. With the notion of expected fraction of time spend in state  $\mathbf{s}$  as  $\mu(\mathbf{s})$  from Sutton et al. [36], this is stated as  $\sum_{\mathbf{s} \in \mathcal{S}_i} \mu(\mathbf{s}) = \frac{1}{|\mathcal{A}|}$ . Let now  $\mathcal{S}_i$  (with  $i \in \{0, 1, \dots, |\mathcal{A}| - 1\}$ ) be an arbitrary state set and  $\mathbf{s}$  an arbitrary state from this set. It must hold, that  $\pi_*(a_i | \mathbf{s}) = 1$ , and consequently  $\pi_*(a_j | \mathbf{s}) = 0$  for all  $i \neq j$ . Hereby, the accuracy  $ACC_{\pi}(\mathcal{E}_{\mathcal{A}})$  denotes the share of selected optimal actions in this environment, following policy  $\pi$ .*

A simple instance of such an environment can be constructed from a `ContextualBandits` scenario. Assume 8 states and 4 actions, where action 0 is optimal for states from  $\mathcal{S}_0 = \{0, 1\}$ , action 1 for states from  $\mathcal{S}_1 = \{2, 3\}$ , action 2 for states from  $\mathcal{S}_2 = \{4, 5\}$ , and action 3 for states from  $\mathcal{S}_3 = \{6, 7\}$ . As the states in the `ContextualBandits` environment are selected uniformly at random in every step, every state set is expectedly visited  $\frac{1}{4}$  of the time. Additionally, the optimal action is different for all state sets, satisfying the conditions from Def. A.2.

**Lemma A.1.** *Let  $\pi$  be any RESTRICTED-SOFTMAX-VQC policy, with w.l.o.g.  $\langle O \rangle_{\mathbf{s}} \in [-1, 1]$ , for all  $\mathbf{s} \in \mathcal{S}$ . Given a uniform environment  $\mathcal{E}_{\mathcal{A}}$ , the performance of the model is upper bounded by*

$$ACC_{\pi}(\mathcal{E}_{\mathcal{A}}) \leq \frac{2}{|\mathcal{A}|} \sum_{k=1}^{|\mathcal{A}|/2} \frac{1}{k}. \quad (\text{A.9})$$

*Proof.* Assume for now, that  $\langle O \rangle_{\mathbf{s}} \in (0, 1]$  for all  $\mathbf{s} \in \mathcal{S}$ . Let  $n = |\mathcal{A}|$  denote the number of actions and the corresponding weights are w.l.o.g. ordered by

$$w_0 \geq w_1 \geq \dots \geq w_{n-2} \geq w_{n-1}. \quad (\text{A.10})$$

The statement  $\pi(a_k | \mathcal{S}_k) \leq \frac{1}{k}$  can be reformulated as  $\sum_{\mathbf{s} \in \mathcal{S}_k} p_{\mathbf{s}} \pi(a_k | \mathbf{s}) \leq \sum_{\mathbf{s} \in \mathcal{S}_k} p_{\mathbf{s}} \frac{1}{k}$ , where  $p_{\mathbf{s}}$  denotes the probability of observing state  $\mathbf{s}$  out of set  $\mathcal{S}_k$ . For the inequality to hold, it is sufficient that  $\pi(a_k | \mathbf{s}) \leq \frac{1}{k}$  for all  $\mathbf{s} \in \mathcal{S}_k$ , which is proven by contradiction:

$$\frac{e^{w_k \langle O \rangle_{\mathbf{s}}}}{\sum_{a'} e^{w_{a'} \langle O \rangle_{\mathbf{s}}}} > \frac{1}{k} \quad (\text{A.11})$$

$$\Leftrightarrow (k-1) \cdot e^{w_k \langle O \rangle_{\mathbf{s}}} > \sum_{a' \neq k} e^{w_{a'} \langle O \rangle_{\mathbf{s}}} \quad (\text{A.12})$$

$$\Leftrightarrow k-1 > \underbrace{\frac{e^{w_0 \langle O \rangle_{\mathbf{s}}}}{e^{w_k \langle O \rangle_{\mathbf{s}}}}}_{\geq 1} + \dots + \underbrace{\frac{e^{w_{n-1} \langle O \rangle_{\mathbf{s}}}}{e^{w_k \langle O \rangle_{\mathbf{s}}}}}_{\geq 1} \geq k-1 \quad \not\leq \quad (\text{A.13})$$

The last step uses Eq. (A.10) combined with the monotonicity of the exponential function. As  $\mathcal{E}_{\mathcal{A}}$  is an uniform environment, for the described policy it holds  $ACC_{\pi}(\mathcal{E}_{\mathcal{A}}) \leq \frac{1}{|\mathcal{A}|} \sum_{k=1}^{|\mathcal{A}|} \frac{1}{k}$ . An improvement of this bound can be achieved by allowing  $\langle O \rangle_{\mathbf{s}} \in [-1, 1]$ . Multiplying with a negative value inverts the inequality chain from Eq. (A.10) to  $-w_0 \leq -w_1 \leq \dots \leq -w_{n-2} \leq -w_{n-1}$ , which introduces the missing factor of 0.5 into Eq. (A.9). Hereby it is implicitly assumed that environment contains an even amount of actions, but it is straightforward to adapt the bound for the odd case. The case  $\langle O \rangle_{\mathbf{s}} = 0$  does not lead to any improvement, as all actions will be selected with equal probability.  $\square$

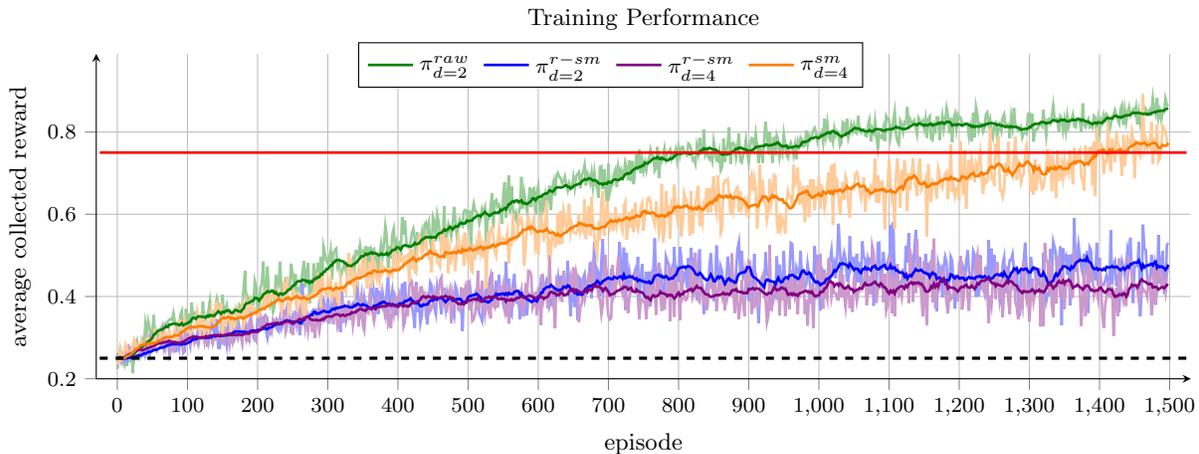


FIG. A.1. Experiments on a `ContextualBandits` environment (which is uniform following Def. A.2) demonstrate the implications of Lem. A.1. The reward structure is defined in a way, such that the expected reward is equivalent to the introduced notion of accuracy  $ACC_{\pi}(\mathcal{E}_A)$ . A `RESTRICTED-SOFTMAX-VQC`  $\pi^{r-sm}$  (with circuit depth  $d = 2$  and  $d = 4$ ) does not perform even close to the theoretical limit of 0.75, indicated by the red horizontal line, while the `RAW-VQC`  $\pi^{raw}$  and `SOFTMAX-VQC`  $\pi^{sm}$  surpass this value. The `RAW-VQC` outperforms the `SOFTMAX-VQC`, even with a shallower circuit. To achieve this, we used the considerations on global policy construction from Sec. IV. All results are averaged over 50 independent experiments.

This upper bound on performance makes the `RESTRICTED-SOFTMAX-VQC` unsuited for uniform environments with large action spaces. In fact, for  $|\mathcal{A}| \rightarrow \infty$  the accuracy converges to 0. Already for 4 actions, the accuracy is bounded by  $\frac{2}{4}(\frac{1}{1} + \frac{1}{2}) = \frac{3}{4}$ , experimental results on the `ContextualBandits` environment described above are depicted in Fig. A.1. We expect this result to hold in a weakened form for more general environments.

### 3. Implications for `SOFTMAX-VQC` Policy

Directly following from Lem. A.1, one can make also a statement about the non-restricted `SOFTMAX-VQC` policy:

**Corollary A.1.** *Let  $\pi$  be any `SOFTMAX-VQC` policy and  $\mathcal{E}_A$  a uniform environment. In order to not impose any constraints following Lem. A.1, at most two actions can be associated with one unique observable. This results in a total requirement of  $\lceil |\mathcal{A}|/2 \rceil$  different observables.*

Following Cor. A.1, the number of measured observables needs to increase linearly with the size of the action space, to avoid a bound on RL performance. Of course, this is only a necessary and not sufficient condition for optimal performance of the model. We assume, that these results at least partially extend to more general environments.

**Corollary A.2.** *In order to not put any performance constraints on the `SOFTMAX-VQC` policy by construction, it has to incorporate  $\mathcal{O}(|\mathcal{A}|)$  different observables  $O_a$ , i.e. the number has to scale linearly in the number of actions.*

Unfortunately, a naive interpretation of Cor. A.2 makes it impossible to circumvent the discussed bad scaling w.r.t. circuit sampling. Strategies to avoid the aforementioned scaling by a suitable choice of observables combined with classical post-processing might exist, but will not be considered in this paper. We are confident that the stated arguments and the performance advantage of the `RAW-VQC` demonstrated in Sec. V provide adequate justification for the focus on this formulation.

## Appendix B: Action Decoding with Local and Global Observables

Continuing with the considerations from Subsec. IV A, for an environment with 2 actions, we have to define  $\mathcal{V}_{a=0}$  and  $\mathcal{V}_{a=1}$ . Following Jerbi et al. [21] for a 3-qubit system yields:

$$\mathcal{V}_{a=0}^{\text{loc}} = \{|000\rangle, |001\rangle, |010\rangle, |011\rangle\} \quad (\text{B.1})$$

$$\mathcal{V}_{a=1}^{\text{loc}} = \{|100\rangle, |101\rangle, |110\rangle, |111\rangle\} \quad (\text{B.2})$$

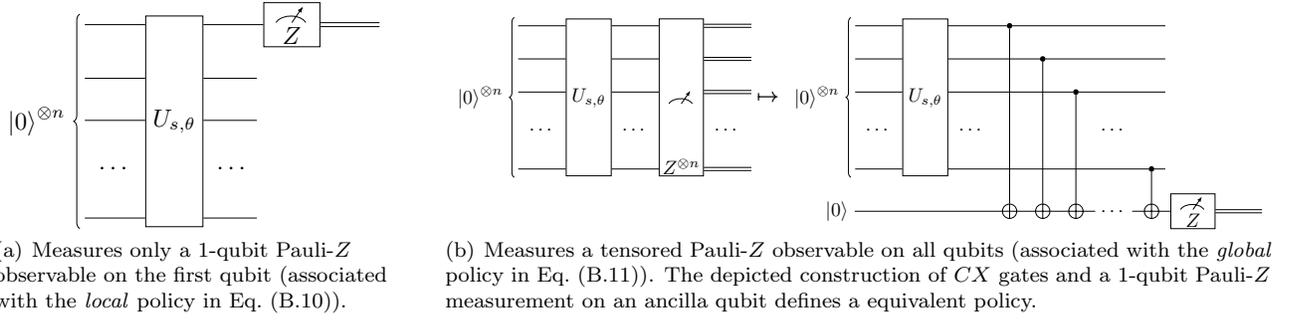


FIG. B.1. Two types of observables, that give rise to different policy formulations.

Looking at the above partitioning, all relevant information seems to be contained in the first qubit. For a measurement in the computational basis of the quantum state prepared by the VQC, that returns the bitstring  $b_2 b_1 b_0$ , the action is given as  $a = b_2$ . In the more general case of  $n$  qubits and 2 actions, this consequently generalizes to  $a^{\text{loc}} \leftarrow b_{n-1}$ , which can also be expressed as

$$\mathcal{V}_a^{\text{loc}} = \{|b_{n-1} b_{n-2} \cdots b_1 b_0\rangle \mid a = b_{n-1}\}. \quad (\text{B.3})$$

At the other extreme, in the case of 2 actions, we can select the action as  $a^{\text{glob}} \leftarrow \bigoplus_{i=0}^{n-1} b_i$ , i.e., apply a parity function. For the sake of completeness, the corresponding eigenstate sets for 3 qubits is

$$\mathcal{V}_{a=0}^{\text{glob}} = \{|000\rangle, |011\rangle, |101\rangle, |110\rangle\} \quad (\text{B.4})$$

$$\mathcal{V}_{a=1}^{\text{glob}} = \{|001\rangle, |010\rangle, |100\rangle, |111\rangle\}, \quad (\text{B.5})$$

and more generally for  $n$  qubits:

$$\mathcal{V}_a^{\text{glob}} = \left\{ |b_{n-1} b_{n-2} \cdots b_1 b_0\rangle \mid a = \bigoplus_{i=0}^{n-1} b_i \right\}. \quad (\text{B.6})$$

So far, we interpreted the policy computation routine as the assignment of the post-measurement state to the containing partition. For the above 2-action special cases in Eqs. (B.3) and (B.6), it is possible to instead model the policy in terms of expectation values of observables:

$$\pi_{\Theta}(a|\mathbf{s}) = \sum_{v \in \mathcal{V}_a} \langle \psi_{\mathbf{s},\Theta} | v \rangle \langle v | \psi_{\mathbf{s},\Theta} \rangle \quad (\text{B.7})$$

$$= \frac{\sum_{v \in \mathcal{V}_a} \langle \psi_{\mathbf{s},\Theta} | v \rangle \langle v | \psi_{\mathbf{s},\Theta} \rangle - (1 - \sum_{v \in \mathcal{V}_{\tilde{a}}} \langle \psi_{\mathbf{s},\Theta} | v \rangle \langle v | \psi_{\mathbf{s},\Theta} \rangle) + 1}{2} \quad (\text{B.8})$$

$$= \frac{\sum_{v \in \mathcal{V}_a} \langle \psi_{\mathbf{s},\Theta} | v \rangle \langle v | \psi_{\mathbf{s},\Theta} \rangle - \sum_{v \in \mathcal{V}_{\tilde{a}}} \langle \psi_{\mathbf{s},\Theta} | v \rangle \langle v | \psi_{\mathbf{s},\Theta} \rangle + 1}{2}, \quad (\text{B.9})$$

where  $\tilde{a}$  denotes the *complement* action of  $a$ .

It is easy to check, that measuring the observable  $Z \otimes I^{\otimes n-1}$  returns the value +1, iff the post-measurement state lives in the space spanned by the elements of  $\mathcal{V}_{a=0}^{\text{loc}}$ , as defined in Eq. (B.3). Vice versa, in all other cases, the measurement outputs a value of -1. This simplifies the above equation to

$$\pi_{\Theta}^{\text{loc}}(a|\mathbf{s}) = \frac{(-1)^a \cdot \langle \psi_{\mathbf{s},\Theta} | Z \otimes I^{\otimes n-1} | \psi_{\mathbf{s},\Theta} \rangle + 1}{2}. \quad (\text{B.10})$$

Returning to the projector formalism from Def. 1, this can alternatively also be expressed as  $\pi_{\Theta}^{\text{loc}}(a|\mathbf{s}) = \langle \psi_{\mathbf{s},\Theta} | \sum_{\mathbf{b} \in \{0,1\}^{n-1}} |a\rangle \langle \mathbf{b}| \langle \mathbf{b}| \psi_{\mathbf{s},\Theta} \rangle$ , i.e., projections onto the two respective sub-spaces for  $a = 0$  and  $a = 1$ .

A similar argument can be made for  $\mathcal{V}_a^{\text{glob}}$ , with the difference that the observable has to be  $Z^{\otimes n}$ , which gives

$$\pi_{\Theta}^{\text{glob}}(a|\mathbf{s}) = \frac{(-1)^a \cdot \langle \psi_{\mathbf{s},\Theta} | Z^{\otimes n} | \psi_{\mathbf{s},\Theta} \rangle + 1}{2}. \quad (\text{B.11})$$

bitstring $\mathbf{b}$	$EI_{f_C}(\mathbf{b})$	containing partition	bitstring $\mathbf{b}$	$EI_{f_C}(\mathbf{b})$	containing partition
0 0 0 0	2	$\mathcal{C}_{a=0}$	1 0 0 0	3	$\mathcal{C}_{a=2}$
0 0 0 1	2	$\mathcal{C}_{a=1}$	1 0 0 1	3	$\mathcal{C}_{a=3}$
0 0 1 0	2	$\mathcal{C}_{a=0}$	1 0 1 0	3	$\mathcal{C}_{a=2}$
0 0 1 1	2	$\mathcal{C}_{a=1}$	1 0 1 1	3	$\mathcal{C}_{a=3}$
0 1 0 0	2	$\mathcal{C}_{a=0}$	1 1 0 0	3	$\mathcal{C}_{a=3}$
0 1 0 1	2	$\mathcal{C}_{a=1}$	1 1 0 1	3	$\mathcal{C}_{a=2}$
0 1 1 0	2	$\mathcal{C}_{a=0}$	1 1 1 0	3	$\mathcal{C}_{a=3}$
0 1 1 1	2	$\mathcal{C}_{a=1}$	1 1 1 1	3	$\mathcal{C}_{a=2}$

TABLE C.1. Extracted information of the post-processing function  $f_C$  for the given partitioning  $\mathcal{C}$  for all 16 bitstrings. The marked bits are used to get an unambiguous assignment to the respective partitions. It is easy to check that one can not go with less information, consequently, the count corresponds to the extracted information.

As above, this could also be stated as projections onto the respective subspaces by reformulating  $\pi_{\Theta}^{\text{glob}}(a | \mathbf{s}) = \langle \psi_{\mathbf{s}, \Theta} | \sum_{\mathbf{b} \in \{0,1\}^n}^{\oplus \mathbf{b}=a} |\mathbf{b}\rangle \langle \mathbf{b}| | \psi_{\mathbf{s}, \Theta} \rangle$ . Note, in general the post-measurement states for both formulations are different.

As also visualized in Fig. B.1, these approaches correspond to using a *local* and *global* observable, respectively. More precisely, for the left diagram, one should refer to a 1-local measurement, as only a 1-qubit observable is measured. For a  $q$ -local measurement, a  $q$ -qubit observable is measured on some subset containing  $q$  out of the  $n$  qubits. When  $q = n$ , we arrive at a global observable, as shown in the right part of the diagram. It seems plausible, that as  $q$  approaches  $n$ , the measurement can be thought of as becoming more and more global. The two setups displayed in Fig. B.1 are only the edge cases. Let us assume a post-processing function that decides on an action based on the parity of the first  $q$  bits of the reconstructed bitstring. This can be expressed as

$$\pi_{\Theta}^{\text{q-loc}}(a | \mathbf{s}) = \frac{(-1)^a \cdot \langle \psi_{\mathbf{s}, \Theta} | Z^{\otimes q} \otimes I^{\otimes n-q} | \psi_{\mathbf{s}, \Theta} \rangle + 1}{2}, \quad (\text{B.12})$$

where  $\pi_{\Theta}^{\text{n-loc}}$  is equivalent to  $\pi_{\Theta}^{\text{glob}}$ . Note that in all case only one bit of information is necessary to select one of the two actions as  $\log_2(2) = 1$ . Still, experiments in Sec. V suggest, that the RL performance benefits from more global measurements.

This type of formulation removes the need to explicitly store partitionings and hence avoids the caveats described above. Unfortunately, this analysis only works for some special cases, i.e., when all the information is compressed into a subset of the qubits. Also, it does not generalize to larger action spaces, as the derivation of Eqs. (B.10) to (B.12) had to assume  $|\mathcal{A}| = 2$ . Apart from that, the distinction between local and global observables in the considerations above is slightly incorrect. Instead of performing a global measurement to evaluate Eq. (B.11), we could get the same result by measuring a 1-qubit Pauli- $Z$  observable on an ancilla qubit, as visualized in the right part of Fig. B.1. This ancilla qubit is initialized to  $|0\rangle$ , and after the evolution of the system with  $U_{\mathbf{s}, \Theta}$ , it interacts via a  $CX$ -gate with each of the  $n$  original qubits.

### Appendix C: Extended Example on Extracted Information and Globality Measure

This appendix deals with a closer analysis of the extracted information and globality measure of the partitioning example introduced in Subsubsec. IV B 1:

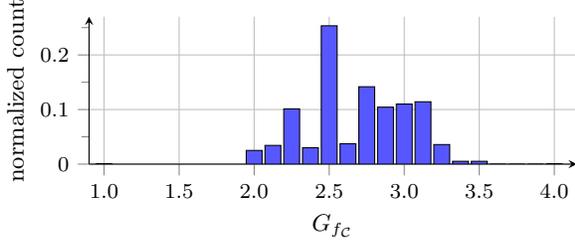
$$\mathcal{C}_{a=0} = \{0000, 0010, 0100, 0110\} \quad (\text{C.1})$$

$$\mathcal{C}_{a=1} = \{0001, 0011, 0101, 0111\} \quad (\text{C.2})$$

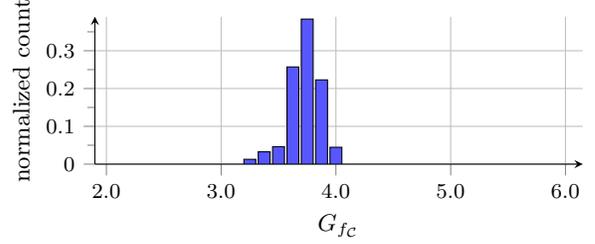
$$\mathcal{C}_{a=2} = \{1000, 1010, 1101, 1111\} \quad (\text{C.3})$$

$$\mathcal{C}_{a=3} = \{1001, 1011, 1100, 1110\} \quad (\text{C.4})$$

As the system is really small, it is straightforward to determine the extracted information following Def. 2 for all  $2^4 = 16$  bitstrings, by just considering all bit combinations. However, as already discussed previously, this is not feasible in the general case.



(a) Histogram for  $n = 4$  and  $|\mathcal{A}| = 2$ . Of all 6435 possible partitionings only one instance exhibits the optimal globality value  $G_{fc} = 4$ .



(b) Histogram for  $n = 6$  and  $|\mathcal{A}| = 4$ . It is basically impossible to guess an partitioning with optimal globality value  $G_{fc} = 6$  from all  $2.8 \cdot 10^{34}$  possibilities.

FIG. D.1. Histogram of globality values over all possible partitionings.

With this work done, it is straightforward to compute the associated globality measure following Eq. (12) as

$$G_{fc} = \frac{1}{2^4} \sum_{\mathbf{b} \in \{0,1\}^4} EI_{fc}(\mathbf{b}) \quad (\text{C.5})$$

$$= \frac{8 \cdot 2 + 8 \cdot 3}{16} = 2.5. \quad (\text{C.6})$$

The result reads itself as that on average 2.5 bit of information is necessary to make an unambiguous action assignment. This is obviously above the minimum value of  $G_{fc}^{\min} = \log_2(4) = 2$ , but well below the optimum of  $G_{fc}^{\max} = n = 4$ . In App. D 3 we demonstrated how a post-processing function with optimal globality measure can be constructed for this setup.

## Appendix D: Supplementary Material on Construction of an Optimal Partitioning

As discussed throughout Subsec. IV B, it is not trivial to come up with a bitstring partitioning, whose associated post-processing function is optimal w.r.t. the globality measure in Eq. (12). However, this property is highly desirable, as it strongly correlates with RL performance, as demonstrated in Subsec. V A.

### 1. Direct Search for an Optimal Post-Processing Function is Infeasible

Unfortunately, the number of possible partitionings is too large to perform any form of unstructured search. In fact, the number increases super-exponentially with the number of qubits  $n$ . To give some proportion, for  $M$  actions and  $N = 2^n$  bitstrings, there are  $N! / \left[ M! \left( \frac{N}{M} \right)^M \right]$  possibilities, where it is assumed that  $M$  is a power of 2, and  $\mathcal{C}$  is split into sets of equal size. As some point of reference, this evaluates to approximately  $2.8 \cdot 10^{34}$  potential partitionings for  $N = 2^6 = 64$  (i.e., a VQC with 6 qubits) and  $M = 4$ , which corresponds to a small quantum system, even for noisy intermediate-scale quantum (NISQ) standards. Lastly, a post-processing function with an underlying random partitioning is very unlikely to have a high globality value close to  $G_{fc} = n$ , as shown in Fig. D.1.

### 2. Proof of Optimality for Proposed Construction

In the main section, we proposed an approach to recursively construct a partitioning, for which the post-processing function is provably optimal w.r.t. the globality measure. For convenience, we restate Lem. 1 below:

**Lemma.** *Let an arbitrary VQC act on an  $n$ -qubit system. The RAW-VQC policy needs to distinguish between  $M := |\mathcal{A}|$  actions, where  $M$  is a power of 2, i.e.,  $m = \log_2(M) - 1 \in \mathbb{N}_0$ . Using the recursive definition from Eqs. (13) to (15),*

one can define

$$\begin{aligned}\pi_{\Theta}^{glob}(a | \mathbf{s}) &= \sum_{v \in \mathcal{C}_{[a]_2}^{(m)}} \langle \psi_{\mathbf{s}, \Theta} | v \rangle \langle v | \psi_{\mathbf{s}, \Theta} \rangle \\ &\approx \frac{1}{K} \sum_{k=0}^{K-1} \delta_{f_{\mathcal{C}^{(m)}}(\mathbf{b}^{(k)})=a},\end{aligned}$$

where  $K$  is the number of shots,  $\mathbf{b}^{(k)}$  is the bitstring observed in the  $k$ -th experiment, and  $\delta$  is an indicator function. The post-processing function associated with this policy is guaranteed to have the highest possible globality measure value  $G_{f_{\mathcal{C}}} = n$ .

*Proof.* The proof uses induction over  $m$ . The base case for  $m = 0$  for Eqs. (14) and (15) is trivial, as it corresponds to the previous considerations from Eqs. (B.6) and (B.11). The induction step  $m \rightarrow m + 1$  needs to consider the two sets, into which  $\mathcal{C}_{[a]_2}^{(m)}$  gets decomposed by inversely applying Eq. (13):

$$\mathcal{C}_{a_m \cdots a_1 0 a_0}^{(m+1)} = \left\{ \mathbf{b} = b_{n-1} b_{n-2} \cdots b_1 b_0 \mid \boxed{\bigoplus_{i=m+1}^{n-1} b_i = a_0} \wedge \mathbf{b} \in \underbrace{\mathcal{C}_{a_m \cdots a_1 (0 \oplus a_0)}^{(m)}}_{[a]_2} \right\} \quad (\text{D.1})$$

$$\mathcal{C}_{a_m \cdots a_1 1 \tilde{a}_0}^{(m+1)} = \left\{ \mathbf{b} = b_{n-1} b_{n-2} \cdots b_1 b_0 \mid \boxed{\bigoplus_{i=m+1}^{n-1} b_i = \tilde{a}_0} \wedge \mathbf{b} \in \underbrace{\mathcal{C}_{a_m \cdots a_1 (1 \oplus \tilde{a}_0)}^{(m)}}_{[a]_2} \right\}, \quad (\text{D.2})$$

where  $\tilde{a}_0$  indicates a bitflip. The property of maximum globality of those two sets w.r.t. to any partition  $\mathcal{C}^{(m+1)}$  is directly transferred by the induction requirement, as the information of all  $m$  least-significant bits is required for that distinction. The marked parts in the above equations highlight, that also the remaining  $n - m$  most-significant bits are required for deciding between action  $[a_m \cdots a_1 0 a_0]_{10}$  and  $[a_m \cdots a_1 1 \tilde{a}_0]_{10}$ . Consequently, it is necessary to always consider the entire bitstring, which implies a globality measure value of  $G_{f_{\mathcal{C}}} = n$ .  $\square$

### 3. Example of Optimal Partitioning

To get some intuition for the construction arising from Lem. 1, we construct a global partitioning for a setup with  $n = 4$  qubits and  $M = 4$  actions. Consequently, the partitions for the respective actions can be derived recursively with  $m = \log_2(4) - 1 = 1$ .

$$\mathcal{C}_{[0]_2=00}^{(1)} = \left\{ \mathbf{b} \mid \boxed{b_3 \oplus b_2 \oplus b_1 = 0} \wedge \mathbf{b} \in \mathcal{C}_{0 \oplus 0 = [0]_2}^{(0)} \right\} = \left\{ \boxed{000} 0, \boxed{011} 0, \boxed{101} 0, \boxed{110} 0 \right\} \quad (\text{D.3})$$

$$\mathcal{C}_{[3]_2=11}^{(1)} = \left\{ \mathbf{b} \mid \boxed{b_3 \oplus b_2 \oplus b_1 = 1} \wedge \mathbf{b} \in \mathcal{C}_{1 \oplus 1 = [0]_2}^{(0)} \right\} = \left\{ \boxed{001} 1, \boxed{010} 1, \boxed{100} 1, \boxed{111} 1 \right\} \quad (\text{D.4})$$

In both cases, after one step of recursion, the base case is reached.

$$\mathcal{C}_{[0]_2}^{(0)} = \{ \mathbf{b} \mid b_3 \oplus b_2 \oplus b_1 \oplus b_0 = 0 \} \quad (\text{D.5})$$

The construction for the remaining two partitions works totally equivalent:

$$\mathcal{C}_{[1]_2=01}^{(1)} = \left\{ \mathbf{b} \mid \boxed{b_3 \oplus b_2 \oplus b_1 = 1} \wedge \mathbf{b} \in \mathcal{C}_{0 \oplus 1 = [1]_2}^{(0)} \right\} = \left\{ \boxed{001} 0, \boxed{010} 0, \boxed{100} 0, \boxed{111} 0 \right\} \quad (\text{D.6})$$

$$\mathcal{C}_{[2]_2=10}^{(1)} = \left\{ \mathbf{b} \mid \boxed{b_3 \oplus b_2 \oplus b_1 = 0} \wedge \mathbf{b} \in \mathcal{C}_{1 \oplus 0 = [1]_2}^{(0)} \right\} = \left\{ \boxed{000} 1, \boxed{011} 1, \boxed{101} 1, \boxed{110} 1 \right\} \quad (\text{D.7})$$

Also here the recursion only has depth one and makes use of the other base case.

$$\mathcal{C}_{[1]_2}^{(0)} = \{ \mathbf{b} \mid b_3 \oplus b_2 \oplus b_1 \oplus b_0 = 1 \} \quad (\text{D.8})$$

We are guaranteed by Lem. 1 that  $G_{f_c} = 4$ , which is also easy to check for this example. It is straightforward to continue from here, i.e. repeatedly split the partitions to account for 8 or 16 actions. For example, the partition for action  $a = 5$  for  $|\mathcal{A}| = 8$  actions is constructed as  $\mathcal{C}_{[5]_2=101}^{(2)} = \left\{ \mathbf{b} \mid b_3 \oplus b_2 = 1 \wedge \mathbf{b} \in \mathcal{C}_{1(0\oplus 1)=[3]_2}^{(1)} \right\} = \{0101, 1001\}$ .

As one can directly observe from this example, the depth of recursion scales only logarithmic in the number of actions. However, the space required for explicit representation of a single partition scales exponentially in  $n - m$ , and overall exponentially in  $n$  for all partitions. Avoiding this caveat, Eq. (18) allows determining the class-association of an arbitrary bitstring w.r.t. the implicitly defined partitioning in an efficient manner. To give a concrete example, let the bitstring of interest be  $\mathbf{b} = 1001$ . We consider an setup with 8 actions, i.e.  $m = \log_2 8 - 1 = 2$ . This gives

$$f_{\mathcal{C}}(\mathbf{b}) = [10(1 \oplus 0)]_{10} = 5, \quad (\text{D.9})$$

which correctly identifies 1001 to be an element of partition  $\mathcal{C}_{[5]_2}^{(2)}$ . As the assignment can be determined efficiently for any bitstring, the post-processing function is efficient.

## Appendix E: Analysis of Fisher Information for Reinforcement Learning Setup

Different metrics for analyzing the *expressibility* and *trainability* for an quantum machine learning (QML) model have recently proposed by Abbas et al. [43]. This work interprets the VQC as a statistical model with the joint distribution  $p_{\Theta}(x, y) = p_{\Theta}(y \mid x)p(x)$  for data pairs  $(x, y)$ . The prior  $p(x)$  describes the distribution of input states, while  $p_{\Theta}(y \mid x)$  gives the relationship between input and output of the model. We need to adapt this notion for the RL setup, which results in  $p_{\pi_{\Theta}}(\mathbf{s}, a) = \pi_{\Theta}(a \mid \mathbf{s})p_{\pi_{\Theta}}(\mathbf{s})$ , where the prior state distribution  $p_{\pi_{\Theta}} : \mathcal{S} \rightarrow [0, 1]$  depends on the policy in most environments. However, for practical reasons, we drop this explicit dependence, while still trying to imitate the distribution of states one would get by following the policy. Due to the loss of statistical independence of data samples (which is one of the most distinguishing features between supervised machine learning (ML) and RL), it is presently unclear to which extent the effective dimension can still be used as a well-defined capacity measure for the function approximation architecture we employ in our work. However, classification and action selection present related tasks. Therefore we assume, that the effective dimension of the VQC circuit architecture (interpreted as a classifier for a supervised ML task) at least serves as a rough indicator of its capacity for policy approximation in the RL context. For the `ContextualBandits` environment, where the consecutive states are sampled independently at random (i.e. independent of the policy), the notion is exact.

The key component of the proposed measures [43] is the Fisher information matrix (FIM)  $F(\Theta) \in \mathbb{R}^{|\Theta| \times |\Theta|}$ . Briefly going into theoretical details, it is a Riemannian metric, given rise to by the Riemannian space formed by  $\Phi$ . From the full parameter space  $\Phi$  each individual parameter set  $\Theta$  is draw. For VQCs consisting mainly of parameterized rotations  $\Phi \subset [-\pi, \pi]^{|\Theta|}$  is a reasonable choice. In practice, it is necessary to approximate the FIM by the empirical FIM. With samples drawn independently and identically distributed from the ground truth  $(s_i, a_i)_{i=1}^k \sim p_{\pi_{\Theta}}(s, a)$ , it is given as

$$\tilde{F}_k(\Theta) = \frac{1}{k} \sum_{i=1}^k [\nabla_{\Theta} \ln p_{\pi_{\Theta}}(s_i, a_i) \nabla_{\Theta} \ln p_{\pi_{\Theta}}(s_i, a_i)^t]. \quad (\text{E.1})$$

An alternate formulation from Sequeira et al. [25] drops the dependence on the prior state distribution. More concretely, this reduces Eq. (E.1) to  $\tilde{F}_k(\Theta) = \frac{1}{k} \sum_{i=1}^k [\nabla_{\Theta} \ln \pi_{\Theta}(a_i \mid s_i) \nabla_{\Theta} \ln \pi_{\Theta}(a_i \mid s_i)^t]$ . However, we decided that keeping the potentially inaccurate prior state information (as dependence on the policy  $\pi_{\Theta}$  is dropped) still should be beneficial compared to not using it at all.

### 1. Expressive Power of the VQC-Model

The FIM  $F(\Theta)$ , and for sufficiently high  $k$  also  $\tilde{F}_k(\Theta)$ , captures the geometry of the parameter space, which allows us to define a measure for the expressibility of a given model. More explicitly, the *effective dimension* [43] quantifies the variety of functions that can be approximated with a given model. For the parameter space  $\Phi \subset \mathbb{R}^{|\Theta|}$ , the effective dimension of the statistical model  $\mathcal{M}_{\Theta}$  associated with the VQC setup can be defined as

$$ed_n(\mathcal{M}_{\Phi}) := 2 \frac{\ln \left( \frac{1}{V_{\Phi}} \int_{\Phi} \sqrt{\det \left( I_{|\Theta|} + \frac{n}{2\pi \ln n} \hat{F}(\Theta) \right)} d\Theta \right)}{\ln \left( \frac{n}{2\pi \ln n} \right)}. \quad (\text{E.2})$$

The FIM  $\hat{F}(\Theta)$  is normalized, such that  $\frac{1}{V_\Phi} \int_\Phi \text{tr} \left( \hat{F}(\Theta) \right) d\Theta = |\Theta|$  holds, where  $V_\Phi := \int_\Phi d\Theta$  denotes the volume of the parameter space. In practice, the (normalized) FIM is replaced with the respective empirical formulation. The parameter  $n$  determines the effective resolution of the parameter space. Although the effective dimension is not guaranteed to increase monotonically with this data size, it is usually the case for ML tasks.

## 2. Trainability of the VQC-Model

The spectrum of the FIM, i.e. its eigenvalue distribution, provides insights into the trainability of a model [43]. In general, trainability profits from an uniform spectrum, while a distorted one is suboptimal. As noted previously, in practice the FIM is replaced with its empirical approximation in Eq. (E.1).

## Appendix F: Supplementary Experiments and Conventions

We now establish some conventions regarding experimental setup and reproducibility. Initially, we experimented with a variety of different hyperparameter settings. Overall, the qualitative observations were quite stable throughout. For the results reported in this paper we fixed most hyperparameters, in order to make results more comparable. However, sometimes slight deviations are necessary to improve performance, which is typical for RL and also quantum reinforcement learning (QRL) [48]. To start with, all experiments on the `CartPole-v0` environment use a learning rate of  $\alpha_\theta = 0.01$  for the variational and  $\alpha_\lambda = 0.1$  for the state scaling parameters. In all other environments, a value of  $\alpha = 0.1$  is used for all parameter sets. A similar distinction is made w.r.t. parameter initialization, where `CartPole-v0` setups select  $\theta \sim \mathcal{N}(0, 0.1)$ , while the base option is always to draw the variational parameters uniformly at random from  $(-\pi, \pi]$ . The state scaling parameters are all initialized to the constant value 1.0. The parameter update is performed using the *Adam* optimizer [49], modified with the *AMSGrad* adjustment [50]. A discount factor of  $\gamma = 0.99$  is used in all cases. No baseline function is used in any of the environments, as performance was found to be sophisticated even without. If not stated differently, the architecture from Fig. 2 with a depth of  $d = 1$  is used, where the number of qubits is adjusted to match the state dimensionality. In order to make RL training curves a bit more stable, the results are usually averaged over ten independent runs. Additionally, the performance is averaged over the last 20 episodes (displayed in darker colors). Some plots also denote the performance of a random agent with a black dashed line and the optimal expected reward with a solid black one.

To support the results from Sec. V, we also conducted experiments for other setups and environments. Basically, the qualitative observations were always consistent with the claims we made, although the peculiarity was sometimes weaker or stronger.

### 1. Increased Quantum Circuit Depth

Instead of using circuits with depth  $d = 1$ , we use data re-uploading with depth  $d = 2$  on the `CartPole-v0` environment. Due to this, and as the resulting circuits contain 40 instead of 24 parameters, the RL performance intuitively should improve. In fact, that is what can be observed in Fig. F.1. Compared to the results in Fig. 3, the convergence speed of the policies  $\pi_{=3}^{Gfc}$  and  $\pi_{=1}^{Gfc}$  has improved quite a bit. Also, the least-global policy is finally able to learn a close to optimal policy after 1000 additional episodes, which was not the case previously. Initially, it actually outperforms the policy with the higher policy but caught up with after approx. 600 episodes. We consider this behavior to be caused by statistical fluctuation, which is common for RL training. As the global policy  $\pi_{=4}^{Gfc}$  showed already a good performance for  $d = 1$ , there was not much room for improvement. Again we back up the results by the respective effective dimensions in Fig. F.1. Here also the predicted pattern holds, with a slight overall improvement over the smaller models.

Overall we conclude, that the increasing model complexity benefits the RL performance and general expressibility of the model for all policy formulations. Still, there is a strong correlation between globality and RL performance, although it is slightly less pronounced than in the original setting. As it is highly desirable for NISQ hardware to keep the circuit complexity as low as possible, using a sophisticated post-processing function should be preferred over increasing the circuit depth.

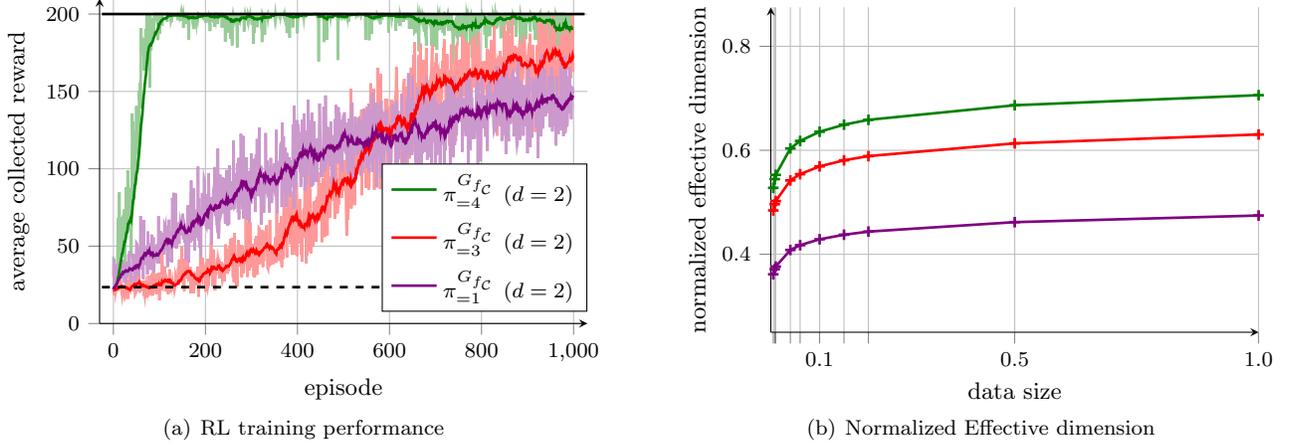


FIG. F.1. RL training performance and associated effective dimension on the `CartPole-v0` environment. The same setup and policy formulations as in Fig. 3 are used, only the depth of the underlying VQCs is increased to  $d = 2$ .

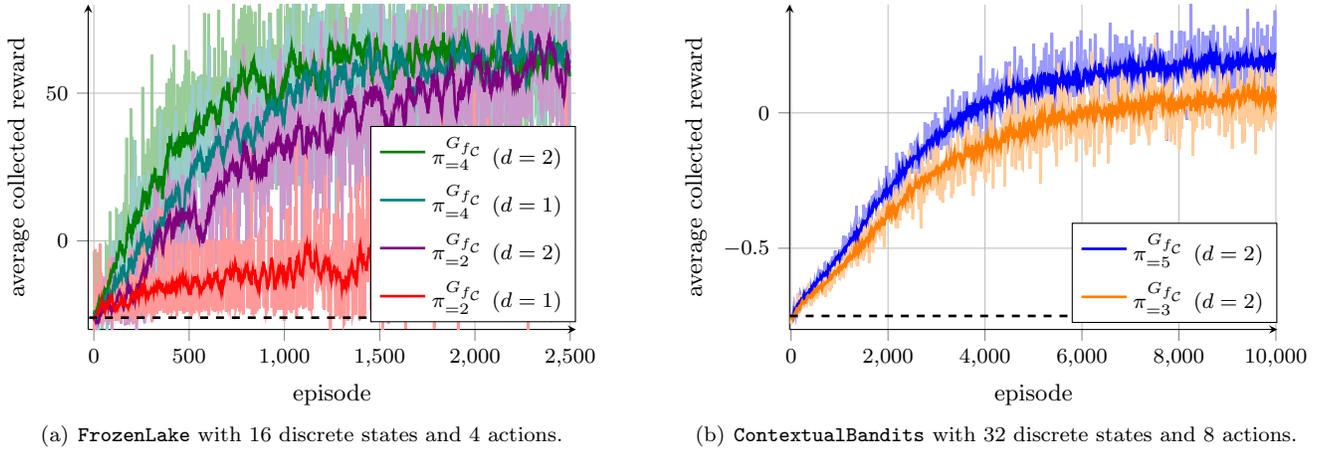


FIG. F.2. RL training performance for different environments. The underlying global post-processing functions are determined as described in Subsubsec. IV B 2. These results are averaged over 100 independent experiments;

		depth $d = 1$	depth $d = 2$	depth $d = 3$	depth $d = 4$
4 qubits ( $ \Theta  = 24, -, -, -$ )	$\pi_{G_{fc}=4}$	0%			
	$\pi_{G_{fc}=3}$	17%			
	$\pi_{G_{fc}=1}$	50%			
6 qubits ( $ \Theta  = 36, 60, -, -$ )	$\pi_{G_{fc}=6}$	0%	0%		
	$\pi_{G_{fc}=3}$	26%	20%		
	$\pi_{G_{fc}=1}$	48%	33%		
8 qubits ( $ \Theta  = 48, 80, 112, -$ )	$\pi_{G_{fc}=8}$	0%	0%	0%	
	$\pi_{G_{fc}=3}$	30%	21%	18%	
	$\pi_{G_{fc}=1}$	38%	26%	25%	
10 qubits ( $ \Theta  = 60, 100, 140, 180$ )	$\pi_{G_{fc}=10}$	1%	0%	0%	0%
	$\pi_{G_{fc}=3}$	28%	27%	19%	16%
	$\pi_{G_{fc}=1}$	30%	28%	23%	20%

TABLE F.1. Percentage of eigenvalues of FIM for two actions and models of increasing complexity that are close to zero ( $< 10^{-7}$ ). The empirical FIM is estimated with 100 random parameter sets for each of the 100 random states  $\mathbf{s}$ . Unlike in Fig. 4, the states elements are sampled uniformly at random from  $[-\pi, \pi)$  to allow for statements abstracted from a concrete RL environment.

## 2. Extension to Other Environments

To really make use of the construction of global post-processing functions for larger action spaces proposed by Lem. 1, we now take a look at two additional environments. The first one is the gridworld environment **FrozenLake** [42], which has to decide between 4 possible actions in every step. Consequently, the lowest possible globality value of a suitable policy is  $G_{fc} = \log_2(4) = 2$ , while an optimal formulation satisfies  $G_{fc} = n = 4$ . The training results for those two policies and different circuit depth is depicted at the top of Fig. F.2. We can basically observe the same pattern as throughout this paper, where a more global post-processing function improves the convergence speed. Also increasing the overall model complexity benefits the performance, wherefore the gap in performance for the different policy formulation decreases.

The second choice is a **ContextualBandits** environment with 32 states and 8 actions. As we encode the states via 1-qubit rotations in a binary fashion, the VQC has  $\log_2(32) = 5$  qubits. This implies an upper bound of  $G_{fc} \leq 5$  and a lower bound of  $G_{fc} \geq 3$ . The performance of the two models is depicted in the lower part of Fig. F.2. Also here the predicted correlation can be observed, although the difference is not that significant. This might be partly down to the reason, that both models struggle to come close to the optimal expected reward of 1.0. By using more sophisticated encoding schemes, or bigger models, one should be able to change this.

We also computed the associated effective dimensions and Fisher spectra, which again followed the predicted scheme. Overall it can be concluded, that the proportionality of the globality measure associated with a post-processing function and its RL performance translates to a variety of environments.

### Appendix G: Abstracted Analysis of Fisher Spectrum

As the analysis of the Fisher information spectrum is a powerful tool to assess trainability, we apply it to a wide range of different setups. In order to not restrict the considerations to a specific environments, we keep things as general as possible by sampling state values uniformly at random from  $[-\pi, \pi]^{|s|}$ . Due to the periodicity of the rotation gates used for encoding, this should cover a wide range of potential scenarios. The most critical property of the Fisher information spectrum is the concentration of eigenvalues close to 0. A high proportion of small eigenvalues indicates a flat parameter space, which makes optimization with any gradient-based technique difficult.

The results for systems ranging from 4 to 10 qubits and depths  $d = 1$  to  $d = 4$  are summarized in Tab. F.1. All experiments assumed an action space of size 2. Most interestingly, the percentage of small eigenvalues for a global post-processing function is almost negligible in all cases. On the contrary, the spectra for policies based on post-processing functions with  $G_C = 1$  are quite degenerated. As one would expect, the post-processing functions with  $G_C = 3$  start out quite well, yet the farther they deviate from the optimal globality value, the more degeneration occurs. These results solidify the statement, that models with global post-processing functions benefit a wide range of applications.

Interestingly, the convergence of eigenvalues towards 0 does not seem to be proportional to the system size. This has some potential implications w.r.t. the barren plateau problem, which is closely related to the trainability of a model. The term describes the observation, that the expectation value and also the variance of the gradients w.r.t. the parameters decrease exponentially with the number of qubits [51]. Abbas et al. [43] relates this to the Fisher information spectrum, i.e. the model is vulnerable to barren plateaus, iff  $\text{tr}(\mathbb{E}_\Theta [F(\Theta)])$  decreases exponentially with increasing system size [51]. Following Tab. F.1, no setup shows a progressive convergence of eigenvalues to 0, although there are some quantitative differences. Probably the considered model sizes are probably still too small for barren plateaus to occur, so for concluding statements additional investigation is necessary. However, following the above statement, barren plateaus are at least unlikely to occur, especially for models with global post-processing functions. This is in line with the results from Abbas et al. [43], where using a complex feature map suppresses barren plateaus. However, the interpretation of the global post-processing function as global measurement somewhat contradicts Cerezo et al. [10]. They stated, that more global measurements enhance the likelihood of barren plateaus in the optimization landscape. Resolving this dissent requires extended experiments with a larger qubit count and circuit depth. Last but not least, it has to be stated, that a problem related to barren plateaus is also known in the classical case. More concretely, big deep neural networks often suffer from vanishing gradients [52]. If the results presented in this section can be extended to larger systems, their advantage in terms of trainability might point towards a possible quantum advantage.