Quantum Policy Gradient Algorithm with Optimized Action Decoding

Nico Meyer [1 2] **Daniel D. Scherer** [1] **Axel Plinge** [1] **Christopher Mutschler** [1] **Michael J. Hartmann** [2]

Abstract

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 an action decoding procedure for a quantum policy gradient approach. We introduce a 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.

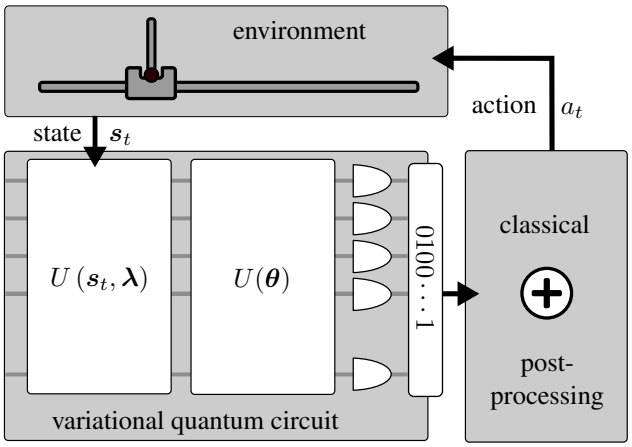


Figure 1. Proposed method: The prepared quantum state is measured in the computational basis. The results are post-processed (with a function maximizing our proposed *globality measure*) before selecting an action. The update of the parameters $\Theta = (\boldsymbol{\theta}, \boldsymbol{\lambda})$ utilizes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

1. 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 potentially has an undesirable sampling and model complexity (Kakade, 2003; Nielsen, 2015; Poggio et al., 2020). Training data is obtained by interaction with the environment via actions, which returns a reward value and a new state. One can optimize the parameters of the policy to maximize the long-term reward with gradient-based techniques, forming a policy gradient (PG) algorithm. Real-world applications can be found e.g. in the domains of self-driving cars (Bojarski et al., 2016) or MIMO beamforming (Maksymyuk et al., 2018).

Exploring the possibilities of other computing paradigms might elevate the impact of RL, e.g. by circumventing the problems caused by increasing parameter complexity of DNN-based 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 (Arute et al., 2019; Zhong et al., 2020; Wu et al., 2021).

The nowadays available quantum devices are considered noisy intermediate-scale quantum (NISQ) hardware (Preskill, 2018), i.e., they only provide a limited amount of qubits that are heavily 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 classically. A promising idea suggests using *variational quantum algorithms* as a platform for quantum machine learning (QML) (Benedetti et al., 2019;

[1]Fraunhofer IIS, Fraunhofer Institute for Integrated Circuits IIS, Nuremberg, Germany [2]Department of Physics, Friedrich-Alexander University Erlangen-Nuremberg (FAU), Erlangen, Germany. Correspondence to: Nico Meyer <nico.meyer@iis.fraunhofer.de>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

Cerezo et al., 2021), for which a certain degree of resilience to the inevitable hardware noise has been reported (Li et al., 2017; Moll et al., 2018; Sharma et al., 2020; Fontana et al., 2021). Variational quantum algorithms use a *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 (Goto et al., 2021; Schuld et al., 2021). For specific problems, variational quantum algorithms are known to exhibit provable quantum advantage (Liu et al., 2021; Sweke et al., 2021).

Contribution. In variational quantum algorithms, it is necessary to extract classical information from the prepared quantum state. Whereas there has been work on obtaining a maximal amount of information about the state via a limited amount of measurements (Huang et al., 2020), 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 for 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 (Jerbi et al., 2021) in Section 3, we start with a formulation in terms of projective measurements in Section 4.1. This is then decomposed into a measurement in the computational basis and the successive application of a classical post-processing function in Section 4.2. Our developed *globality measure* allows to compare specific instances of those functions. Furthermore, 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., our procedure is feasible for problems with large action spaces.

We observe a strong correlation between RL performance and our globality measure in Section 5.1 in the RL environments `CartPole`, `FrozenLake`, and different configurations of `ContextualBandits`. 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 Section 5.2. 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 Section 5.3.

2. Related Work

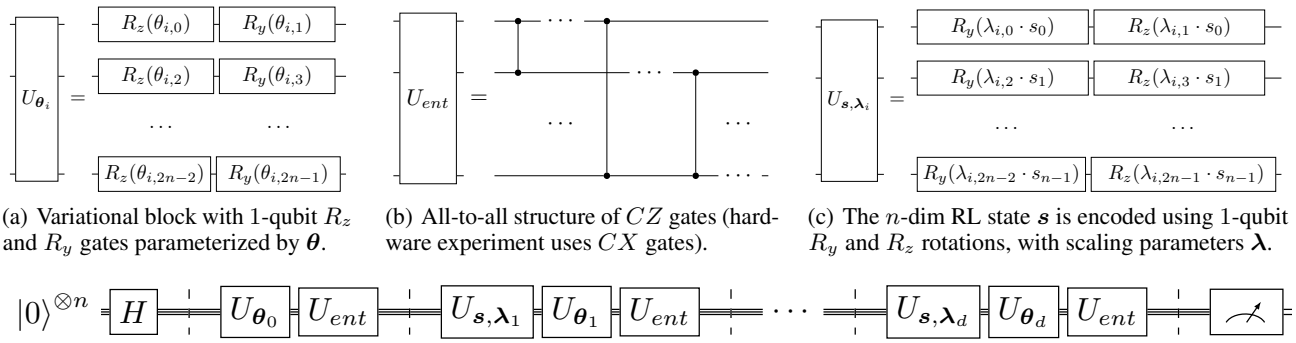
A summary of the current body of work on QRL can be found in Meyer et al. (Meyer et al., 2022). Specific QRL routines have already been realized experimentally (Saggio et al., 2021). An early instance of VQC-based QRL proposes to use a VQC as an approximator for the action-value function (Chen et al., 2020). We follow the lines of Jerbi et al., which uses the VQC for policy approximation, forming a QPG algorithm. Additional work on the QPG approach include an extension to quantum environments (Sequeira et al., 2022), and a modified parameter update to reduce sampling complexity (Meyer et al., 2023). The ideas of value-function and policy approximation are combined into actor-critic approaches (Wu et al., 2020; Kwak et al., 2021), which also can benefit from our contribution.

While the algorithmic routine of QPG follows the idea of classical PG, the design of the VQC function approximator is an ongoing research field. The typical architecture features three different 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 (Pérez-Salinas et al., 2020; Schuld et al., 2021; Periyasamy et al., 2022) and variational layers (Sim et al., 2019; Kandala et al., 2017), based on the specific problem type. We focus on the necessary measurements, which require special attention in the context of quantum information theory (Braginsky et al., 1995; Nielsen & Chuang, 2010), and also QML with VQCs (Schuld & Petruccione, 2018; Cerezo et al., 2021; Schuld, 2021). 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.

3. 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 $(\mathcal{S}, \mathcal{A}, \mathcal{R}, p, \gamma)$, where $\mathcal{S}$ is the state set, $\mathcal{A}$ is the set of available actions, $\mathcal{R} \subset \mathbb{R}$ is the reward space, $p : \mathcal{S} \times \mathcal{R} \times \mathcal{S} \times \mathcal{A} \to [0, 1]$ describes the environment dynamics, and $0 \leq \gamma \leq 1$ is a discount factor. At each timestep t, the agent observes the environment state $\boldsymbol{s}_t$, and decides on an action a_t. This decision is sampled from the current policy $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$, which defines a probability density function (PDF) over all possible actions a for a given state $\boldsymbol{s}$. The selected action is executed, and the agent receives a scalar reward $r_t \in \mathcal{R}$, after which the environment transitions to state s_{t+1}, following its dynamics p. The objective is to learn a policy, which maximizes the (discounted) return $G_t := \sum_{t'=t}^{H-1} \gamma^{t'-t} \cdot r_{t'}$ for some horizon $H < \infty$ (Sutton & Barto, 2018).

Our work follows the hybrid QPG algorithm proposed by Jerbi et al.. The approach is inspired by the classical `REINFORCE` idea with function approximation (Sutton



(a) Variational block with 1-qubit R_z and R_y gates parameterized by $\boldsymbol{\theta}$.

(b) All-to-all structure of CZ gates (hardware experiment uses CX gates).

(c) The n-dim RL state $\boldsymbol{s}$ is encoded using 1-qubit R_y and R_z rotations, with scaling parameters $\boldsymbol{\lambda}$.

Figure 2. Hardware-efficient quantum circuit, adapted from Jerbi et al.. The parameters are summarized in $\Theta = (\boldsymbol{\theta}, \boldsymbol{\lambda})$. Depending on circuit depth d, the encoding blocks $U_{\boldsymbol{s},\boldsymbol{\lambda}}$, combined with variational blocks $U_{\boldsymbol{\theta}}$ and entanglement blocks U_{ent}, are repeated (i.e. for $d \geq 2$ data re-uploading (Pérez-Salinas et al., 2020) is used). Measurements are performed in the computational basis.

et al., 1999), also referred to as *vanilla policy gradient*. Its centerpiece is the parameterized policy $\pi_\Theta : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$, where Θ 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 α, and some scalar performance measure $J(\Theta)$. The *policy gradient theorem* (Sutton et al., 1999) states the gradient of the performance measure as

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

In practice, the expectation value in Equation (1) is approximated by averaging over several trajectories τ (i.e., sequences of current state, executed action, and received reward for some timesteps), that are generated by following the current policy π_Θ. For a DNN, the gradient of the log-policy w.r.t. the parameters can be obtained using backpropagation (Rumelhart et al., 1986). For VQCs executed on quantum hardware one typically resorts to the parameter-shift rule (Mitarai et al., 2018; Schuld et al., 2019) – as in this paper – or simultaneous perturbation stochastic approximations (SPSA) (Spall, 1998; Wiedmann et al., 2023).

3.1. 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 (Jerbi et al., 2021). The VQC acts on $|0\rangle^{\otimes n}$ (where $|0\rangle$ denotes the 1-qubit computational zero state) with the unitary $U_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}}$, which prepares the quantum state $|\psi_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}}\rangle$. We introduce only the fundamentals of QC and refer the interested reader to Nielsen & Chuang for more details.

Similar to Jerbi et al. we use the hardware-efficient ansatz in Figure 2. Besides the variational parameters $\boldsymbol{\theta}$, there are also trainable scaling parameters $\boldsymbol{\lambda}$ to enhance the expressivity of the model. For ease of notation we denote $\Theta = (\boldsymbol{\theta}, \boldsymbol{\lambda})$.

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

3.2. Reformulation of the RAW-VQC Policy

Jerbi et al. define the RAW-VQC and the SOFTMAX-VQC and suggest, that the latter formulation is superior in terms of RL performance. However, we argue in Appendix A, that is has several drawbacks w.r.t. circuit sampling complexity. We experimentally demonstrate in Section 5, 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 3.1 (RAW-VQC)**.** Given a VQC acting on n qubits, taking as input an RL state $\boldsymbol{s} \in \mathbb{R}^n$, rotation angles $\boldsymbol{\theta} \in [-\pi, \pi]^{|\boldsymbol{\theta}|}$, scaling parameters $\boldsymbol{\lambda} \in \mathbb{R}^{|\boldsymbol{\lambda}|}$, with $\Theta = (\boldsymbol{\theta}, \boldsymbol{\lambda})$, such that it produces the quantum state $|\psi_{\boldsymbol{s},\Theta}\rangle = U_{\boldsymbol{s},\Theta} |0\rangle^{\otimes n}$, we define the RAW-VQC policy:

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

where $\langle P_a \rangle_{\boldsymbol{s},\Theta} = \langle \psi_{\boldsymbol{s},\Theta} | P_a | \psi_{\boldsymbol{s},\Theta} \rangle$ 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 $\mathcal{V} = \{|v_0\rangle, |v_1\rangle, \cdots, |v_{2^n-1}\rangle\}$ is the set of eigenstates of an observable

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

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 slightly more restrictive than the original one, due to the explicit designation of the eigenvalues in Equation (3). This ensures, that by measuring eigenvalue i, we can directly conclude the post-measurement state to be $|v_i\rangle$. However, Definition 3.1 is completely equivalent in terms of all considerations and experiments carried out in Jerbi et al..

4. Analysis of Action Decoding

We now focus on the action decoding scheme of the `RAW-VQC` policy. Definition 3.1 is instantiated such that the observable in Equation (3) can be efficiently replaced with only 1-qubit Pauli operators. We start by measuring a 1-qubit Pauli observable σ_{n-1} on the uppermost qubit. The observed result is one of the eigenvalues $\mu_0 = +1$ or $\mu_1 = -1$, which is interpreted as the bit value $b_{n-1} = \frac{1-\mu}{2}$. 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 σ_{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 Equation (3). We follow the convention that the most significant bit corresponds to the uppermost wire of Figure 2.

4.1. Partitioning of Computational Basis States

Since measurements can typically only be done in the energy eigenbasis of a qubit, we select the Pauli operators σ_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 $\mathcal{V} = \{|0\cdots00\rangle, |0\cdots01\rangle, \cdots, |1\cdots11\rangle\}$. However, our techniques can also be applied to more general combinations of Pauli operators, as all share eigenvalues $+1$ and -1. Following Definition 3.1 this set has to be partitioned, i.e., $\mathcal{V}_a = \{|a_0\rangle, |a_1\rangle, \cdots\} \subseteq \mathcal{V}$ for action a. Let the prepared state be represented in the computational basis (using decimal notation) as $|\psi_{\boldsymbol{s},\Theta}\rangle = c_0|0\rangle + c_1|1\rangle + \cdots + c_{N-1}|N-1\rangle$, with $N = 2^n$. This allows the reformulation of Equation (2) in terms of the absolute squared amplitudes of the prepared quantum state:

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

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

Consequently, (as only Pauli observables are considered) it is sufficient to sum up the absolute squared amplitudes associated with the respective basis states to determine the policy. On quantum hardware 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 selects an action according to the current policy $\pi_\Theta(a|s)$. Starting from Equation (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 $\nabla_\Theta \ln \pi_\Theta(a|\boldsymbol{s}) = \nabla_\Theta \langle P_a\rangle_{\boldsymbol{s},\Theta} \,/\, \langle P_a\rangle_{\boldsymbol{s},\Theta}$, for a trajectory of concrete instances of $\boldsymbol{s}$ and a. To estimate $\langle P_a\rangle_{\boldsymbol{s},\Theta}$ (and also $\nabla_\Theta \langle P_a\rangle_{\boldsymbol{s},\Theta}$), 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 (Jerbi et al., 2021). For larger systems, storing all $\mathcal{V}_a$ is infeasible, as the number of elements scales exponentially in n, independently of quantum or classical hardware (for $n = 64$ qubits, there are 2^{64} possible bit strings, which would require $2^{64}\cdot\log_2(64)$ bit ≈ 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 strongly depends 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 correlates with RL performance.

4.2. Action Decoding with Classical Post-Processing Function

We denote the set of all bitstrings as $\boldsymbol{b} = b_{n-1}b_{n-2}\cdots b_1b_0$ by $\mathcal{C}$, and partition it into disjoint, action-associated sets $\mathcal{C}_a$. We define a *classical post-processing function* $f_\mathcal{C} : \{0,1\}^n \to \{0,1,\cdots,|\mathcal{A}|-1\}$, such that $f_\mathcal{C}(\boldsymbol{b}) = a$, iff $\boldsymbol{b} \in \mathcal{C}_a$ for an partitioning of $\mathcal{C}$. We can reformulate Equation (5):

$$\pi_\Theta(a \mid \boldsymbol{s}) = \sum_{\boldsymbol{b}\in\{0,1\}^n}^{f_\mathcal{C}(\boldsymbol{b})=a} \langle\psi_{\boldsymbol{s},\Theta} \mid \boldsymbol{b}\rangle\langle\boldsymbol{b} \mid \psi_{\boldsymbol{s},\Theta}\rangle \tag{6}$$

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

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

4.2.1. 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 $\boldsymbol{b}$:

Definition 4.1 (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, $\boldsymbol{b} = b_{n-1}b_{n-2}\cdots b_1 b_0$ denotes an arbitrary bitstring. The extracted information $EI_{f_{\mathcal{C}}}(\boldsymbol{b}) \in \mathbb{N}$ is the minimum number of bits b_i necessary, to compute $f_{\mathcal{C}}(\boldsymbol{b})$, i.e. assign $\boldsymbol{b}$ unambiguously to a set $\mathcal{C}_a$.

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

$$\mathcal{C}_{a=0} = \{0000,\ 0010,\ 0100,\ 0110\}, \tag{8}$$
$$\mathcal{C}_{a=1} = \{0001,\ 0011,\ 0101,\ 0111\}, \tag{9}$$
$$\mathcal{C}_{a=2} = \{1000,\ 1010,\ 1101,\ 1111\}, \tag{10}$$
$$\mathcal{C}_{a=3} = \{1001,\ 1011,\ 1100,\ 1110\}. \tag{11}$$

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 $EI_{f_{\mathcal{C}}}(0111) = 2$.

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

$$G_{f_{\mathcal{C}}} := \frac{1}{2^n} \sum_{\boldsymbol{b}\in\{0,1\}^n} EI_{f_{\mathcal{C}}}(\boldsymbol{b}). \tag{12}$$

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 Appendix B.

A lower bound to $G_{f_{\mathcal{C}}}$ is intuitively given by $G_{f_{\mathcal{C}}} \geq \log_2(|\mathcal{A}|)$. The measure is trivially upper bounded by $G_{f_{\mathcal{C}}} \leq n$, which is in line with Holevo's theorem (Holevo, 1973; Nielsen & Chuang, 2010), in that no more than n bits of classical information can be extracted from a n-qubit system in a single measurement. Evaluating Equation (12) for the example above gives $G_{f_{\mathcal{C}}} = 2.5$, i.e., on average 2.5 bits of information are necessary (see Appendix C for the exact computation).

Evaluating Equation (12) explicitly is infeasible for large n, as it requires averaging over 2^n elements. Furthermore, we are not aware of an efficient routine that determines the extracted information for an arbitrary bitstring. Nonetheless, one can define a post-processing function, which has maximal globality according to Equation (12). We discuss this construction in the next section.

4.2.2. Constructing an Optimal Post-Processing Function

As we demonstrate in Section 5, 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 Appendix 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 $G_{f_{\mathcal{C}}} = n$. The set of bitstrings $\boldsymbol{b} = b_{n-1}\cdots b_0$ associated with action a is recursively defined as

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

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:

$$\mathcal{C}^{(0)}_{[0]_2} = \left\{ \boldsymbol{b} \mid \bigoplus_{i=0}^{n-1} b_i = 0 \wedge \boldsymbol{b} \in \{0,1\}^n \right\} \tag{14}$$
$$\mathcal{C}^{(0)}_{[1]_2} = \left\{ \boldsymbol{b} \mid \bigoplus_{i=0}^{n-1} b_i = 1 \wedge \boldsymbol{b} \in \{0,1\}^n \right\} \tag{15}$$

The construction in Equation (13) thus recursively splits Equations (14) and (15) by computing parity values of substrings, until the required number of groups is formed.

Lemma 4.2. *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 Equations (13) to (15) we define*

$$\pi^{glob}_{\Theta}(a \mid \boldsymbol{s}) = \sum_{v\in\mathcal{C}^{(m)}_{[a]_2}} \langle \psi_{\boldsymbol{s},\Theta} \mid v\rangle \langle v \mid \psi_{\boldsymbol{s},\Theta}\rangle \tag{16}$$
$$\approx \frac{1}{K} \sum_{k=0}^{K-1} \delta_{f_{\mathcal{C}^{(m)}}(\boldsymbol{b}^{(k)})=a} \tag{17}$$

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

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

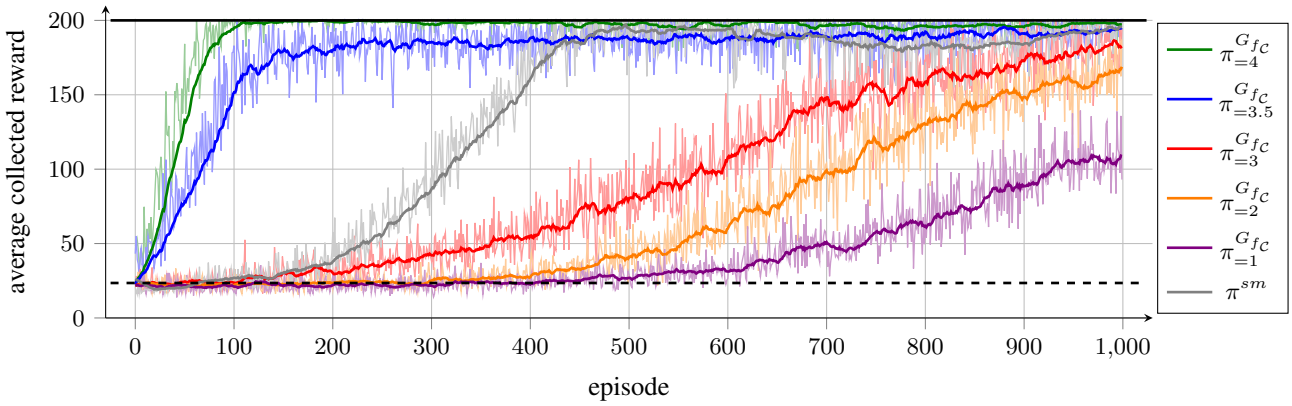


Figure 3. RL training performance of `RAW`-VQC policies with different post-processing functions on the `CartPole-v0` environment, averaged over 10 independent runs and 20 preceding timesteps (dark curves). A higher globality value correlates with a faster convergence to the optimal collected reward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

By construction of the recursive definition the action associated with a specific bitstring $f_{\mathcal{C}}(\boldsymbol{b})$ is given as

$$f_{\mathcal{C}}(\boldsymbol{b}) = \left[b_0 \cdots b_{m-1} \left(\bigoplus_{i=m}^{n-1} b_i \right) \right]_{10} \tag{18}$$

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.

5. 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`, 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 Sections 5.1 and 5.2 focus on the `CartPole` environment, while Section 5.3 and Appendix F also consider `ContextualBandits` and `FrozenLake`, respectively. We establish conventions regarding the experimental setup and reproducibility in Appendix F.

5.1. RL Performance vs. Globality Measure

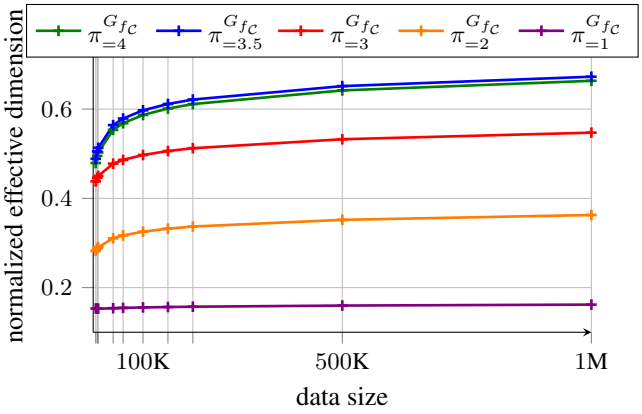
The main experiment is conducted on the `CartPole-v0` environment (Brockman et al., 2016) with a horizon of 200 time-steps. The state space has a dimensionality of 4, with all values scaled to be within $[-1, 1)$. The agent can take two actions, i.e., $|\mathcal{A}| = 2$.

All experiments in Figure 3 (apart from the `SOFTMAX`-VQC policy π^{sm}, which uses a tensored Pauli-Z measurement on all qubits (Jerbi et al., 2021)) use the same architecture, only the post-processing function is modified. The two extreme cases are $G_{fc} = 4.0$, constructed following Lemma 4.2, 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 $\mathcal{C}^{3.5}_{a=0} = \{1,\ 3,\ 5,\ 6,\ 9,\ 10,\ 12,\ 15\}$ and $\mathcal{C}^{3.5}_{a=1} = \{0,\ 2,\ 4,\ 7,\ 8,\ 11,\ 13,\ 14\}$ is used.

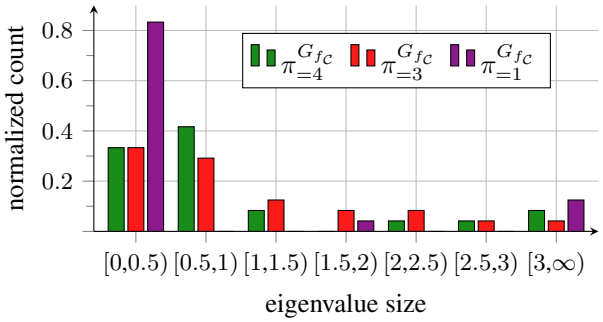
Throughout all considerations, the RL performance clearly benefits from a higher globality value of the underlying post-processing function. The convergence speed is improved, for example, the strategy learned by an agent with an underlying global post-processing function reaches optimal behavior after 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 Appendix F.1). However, as circuit depth is very critical for NISQ devices, using optimal classical post-processing functions is crucial.

Figure 3 also depicts the performance of a `SOFTMAX`-VQC policy (Jerbi et al., 2021). Interestingly, it performs better than `RAW`-VQC with a globality value ≤ 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 Appendix F.2.



(a) Normalized effective dimension



(b) Fisher information spectrum

Figure 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 $\boldsymbol{s}$. We draw the elements of each state from $\mathcal{N}(0, 0.5)$, which mimics the prior state distribution of the `CartPole` environment.

5.2. 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., and a comparative study of various quantum neural network architectures has been conducted in Wilkinson & Hartmann. An adaption of those concepts to the RL setup is deferred to Appendix E.

The expressibility of a model can be quantified using the *effective dimension* (Abbas et al., 2021), 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 Figure 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 Figure 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.

Insights into the trainability of a model are possible by considering the spectrum of the FIM (Abbas et al., 2021), which captures the geometry of the parameter space. Trainability profits from a uniform spectrum, while distorted spectra are suboptimal. Figure 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 Appendix 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.

5.3. Training on Quantum Hardware

To emphasize the practical relevance of our method, we conducted a 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 (Chen et al., 2020; Hsiao et al., 2022), where the trained models model 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 (Sutton & Barto, 2018) for the experiment, which can be implemented with a 3-qubit system. The employed hardware backend is the 5-qubit device `ibmq_manila v1.1.4` (IBM Quantum, 2023). We slightly adapted the VQC architecture and typical RL feedback loop to make hardware usage feasible. First, we replaced the CZ gates from Figure 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

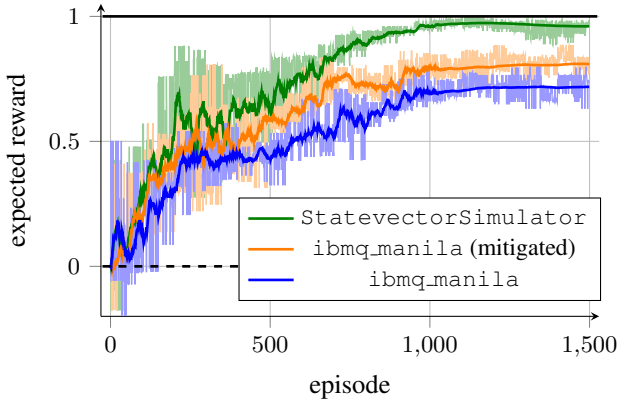


Figure 5. Training performance of a RAW-VQC policy with maximum globality value on an 8-state and 2-action ContextualBandits environment. Two training runs are executed on quantum hardware (wo/ and w/ error mitigation).

Table 1. Test results on the ibmq_manila hardware device with error mitigation (orange curve in Figure 5). As for ContextualBandits the states are sampled uniformly at random, the optimal action is selected in approx. 92% of all cases.

	$s=0$	$s=1$	$s=2$	$s=3$
optimal action	$a=0$	$a=0$	$a=1$	$a=1$
$\pi(a=0 \mid s)$	**0.93**	**0.91**	0.09	0.06
$\pi(a=1 \mid s)$	0.07	0.09	**0.91**	**0.94**
	$s=4$	$s=5$	$s=6$	$s=7$
optimal action	$a=0$	$a=0$	$a=1$	$a=1$
$\pi(a=0 \mid s)$	**0.94**	**0.91**	0.09	0.07
$\pi(a=1 \mid s)$	0.06	0.09	**0.91**	**0.93**

trajectories. Consequently the gradients, and therefore also the parameter updates, are only computed for each 50th time-step. Still, for a horizon of $1,500$ episodes, this adds up to overall $14,640$ expectation values that need to be estimated. With $1,024$ shots to estimate each one, close to 15M circuits had to be evaluated per training run.

The training performance for different setups is displayed in Figure 5. We compare results on the hardware with and without matrix-free measurement error mitigation (Qiskit contributers, 2023; Nation et al., 2021). This is compared to results obtained from noise-free simulation on classical hardware. We also experimented with qiskit noise model instantiated with parameters sampled from the ibmq_manila device – the results were almost identical to the actual hardware. The noise-free simulation clearly produces the best results and is able to learn an basically optimal policy. While this is not the case for the experiments on hardware, there is still a clear improvement over the initial random policy. Hereby, as expected, the mitigated experiment (execution time approx. 360 minutes over two Qiskit Sessions) improves upon the non-mitigated one (execution time approx. 150 minutes in a single Qiskit Session). Interestingly, the performance of all three agents seems to saturate after about 1000 episodes.

The testing results in Table 1 clearly show that the (error-mitigated) hardware-trained 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 noise mainly induced by entangling gates ε_{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. It has to be noted that the overall length of the transpiled circuits stays approximately constant throughout all episodes. The re-calibration of the system after episode 750 of the error-mitigated experiment (reducing ε_{CX} from 0.70% to 0.64%) did not cause a clear change in performance. We assume, that the convergence of the hardware agents towards a non-optimal expected reward is mainly caused by decoherence noise. To improve upon this, one can potentially use an architecture more adapted to the basis gate set and connectivity structure. Apart from that, more advanced error mitigation strategies (Giurgica-Tiron et al., 2020; Mari et al., 2021) could be a suitable option.

6. Conclusion

This paper analyzed the action decoding procedure of the quantum policy gradient (QPG) algorithm originally proposed by Jerbi et al.. 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 model complexity measures. We provided a routine to implement a post-processing function that is optimal with respect to this measure – which is also feasible for large action spaces. Compared to the original RAW-variational quantum circuit (VQC), as well as the SOFTMAX-VQC policy (Jerbi et al., 2021), we achieve significant RL performance improvements, with only negligible classical overhead. With this enhanced QPG algorithm, we are able to execute the entire RL training and testing routine on actual quantum hardware.

Our work focused on RL routines, but in principle our findings can be extended to the realm of supervised and unsupervised learning. More concretely, the post-processing function for action selection can be reformulated to return the labels of a classification problem – with reliable statements on transferability certainly requiring additional experiments. While we did not explicitly claim quantum advantage, the idea of constructing an environment based on the discrete logarithm from Jerbi et al. 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 via the project BayQS 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 "Quromorphic" 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.com/NicoMeyer/qpg_classicalpp. Further information and data is available upon reasonable request.

References

Abbas, A., Sutter, D., Zoufal, C., Lucchi, A., Figalli, A., and Woerner, S. The power of quantum neural networks. *Nat. Comput. Sci.*, 1(6):403–409, 2021. doi: 10.1038/s43588-021-00084-1.

Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J. C., Barends, R., Biswas, R., Boixo, S., Brandao, F. G., Buell, D. A., et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019. doi: 10.1038/s41586-019-1666-5.

Benedetti, M., Lloyd, E., Sack, S., and Fiorentini, M. Parameterized quantum circuits as machine learning models. *Quantum Sci. Technol.*, 4(4):043001, 2019. doi: 10.1088/2058-9565/ab4eb5.

Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., and Zhang, J. End to end learning for self-driving cars. *arXiv:1604.07316*, 2016. doi: 10.48550/arXiv.1604.07316.

Braginsky, V. B., Braginski, V. B., and Khalili, F. Y. *Quantum measurement*. Cambridge University Press, 1995. doi: 10.1017/CBO9780511622748.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. OpenAI Gym. *arXiv:1606.01540*, 2016. doi: 10.48550/arXiv.1606.01540.

Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S. C., Endo, S., Fujii, K., McClean, J. R., Mitarai, K., Yuan, X., Cincio, L., et al. Variational quantum algorithms. *Nat. Rev. Phys.*, 3(9):625–644, 2021. doi: 10.1038/s42254-021-00348-9.

Chen, S. Y.-C., Yang, C.-H. H., Qi, J., Chen, P.-Y., Ma, X., and Goan, H.-S. Variational Quantum Circuits for Deep Reinforcement Learning. *IEEE Access*, 8:141007–141024, 2020. doi: 10.1109/ACCESS.2020.3010470.

Fontana, E., Fitzpatrick, N., Ramo, D. M., Duncan, R., and Rungger, I. Evaluating the noise resilience of variational quantum algorithms. *Phys. Rev. A*, 104(2):022403, 2021. doi: 10.1103/PhysRevA.104.022403.

Franz, M., Wolf, L., Periyasamy, M., Uftrecht, C., Scherer, D. D., Plinge, A., Mutschler, C., and Mauerer, W. Uncovering instabilities in variational-quantum deep Q-networks. *J. Franklin Inst.*, 2022. doi: 10.1016/j.jfranklin.2022.08.021.

Giurgica-Tiron, T., Hindy, Y., LaRose, R., Mari, A., and Zeng, W. J. Digital zero noise extrapolation for quantum error mitigation. In *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 306–316, 2020. doi: 10.1109/QCE49297.2020.00045.

Goto, T., Tran, Q. H., and Nakajima, K. Universal Approximation Property of Quantum Machine Learning Models in Quantum-Enhanced Feature Spaces. *Phys. Rev. Lett.*, 127(9):090506, 2021. doi: 10.1103/PhysRevLett.127.090506.

Hochreiter, S. The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions. *Int. J. Uncertain. Fuzz.*, 6(02):107–116, 1998. doi: 10.1142/S0218488598000094.

Holevo, A. S. Bounds for the Quantity of Information Transmitted by a Quantum Communication Channel. *Problems Inform. Transmission*, 9(3):177–183, 1973. URL https://www.mathnet.ru/php/archive.phtml?wshow=paper&jrnid=ppi&paperid=903&option_lang=eng.

Hsiao, J.-Y., Du, Y., Chiang, W.-Y., Hsieh, M.-H., and Goan, H.-S. Unentangled quantum reinforcement learning agents in the OpenAI Gym. *arXiv:2203.14348*, 2022. doi: 10.48550/arXiv.2203.14348.

Huang, H.-Y., Kueng, R., and Preskill, J. Predicting many properties of a quantum system from very few measurements. *Nat. Phys.*, 16(10):1050–1057, 2020. doi: 10.1038/s41567-020-0932-7.

IBM Quantum. Qiskit Runtime Service, Sampler primitive (version 0.9.1). https://quantum-computing.ibm.com/, 2023.

Jerbi, S., Gyurik, C., Marshall, S., Briegel, H., and Dunjko, V. Parametrized Quantum Policies for Reinforcement Learning. *Adv. Neural Inf. Process. Syst.*, 34:28362–28375, 2021. doi: 10.5281/zenodo.5833370.

Kakade, S. M. *On the Sample Complexity of Reinforcement Learning*. PhD thesis, University College London, 2003. URL https://homes.cs.washington.edu/~sham/papers/thesis/sham_thesis.pdf.

Kandala, A., Mezzacapo, A., Temme, K., Takita, M., Brink, M., Chow, J. M., and Gambetta, J. M. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242–246, 2017. doi: 10.1038/nature23879.

Kashif, M. and Al-Kuwari, S. The impact of cost function globality and locality in hybrid quantum neural networks on NISQ devices. *Machine Learning: Science and Technology*, 4(1):015004, 2023. doi: 10.1088/2632-2153/acb12f.

Kingma, D. P. and Ba, J. Adam: A Method for Stochastic Optimization. *Proc. of the Int. Conf. on Learning Representations (ICLR)*, 2015. doi: 10.48550/arXiv.1412.6980.

Kwak, Y., Yun, W. J., Jung, S., Kim, J.-K., and Kim, J. Introduction to Quantum Reinforcement Learning: Theory and PennyLane-based Implementation. In *International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 416–420, 2021. doi: 10.1109/ICTC52510.2021.9620885.

Li, J., Yang, X., Peng, X., and Sun, C.-P. Hybrid Quantum-Classical Approach to Quantum Optimal Control. *Phys. Rev. Lett.*, 118(15):150503, 2017. doi: 10.1103/PhysRevLett.118.150503.

Liu, Y., Arunachalam, S., and Temme, K. A rigorous and robust quantum speed-up in supervised machine learning. *Nat. Phys.*, 17(9):1013–1017, 2021. doi: 10.1038/s41567-021-01287-z.

Maksymyuk, T., Gazda, J., Yaremko, O., and Nevinskiy, D. Deep learning based massive MIMO beamforming for 5G mobile network. In *International Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS)*, pp. 241–244, 2018. doi: 10.1109/IDAACS-SWS.2018.8525802.

Mari, A., Shammah, N., and Zeng, W. J. Extending quantum probabilistic error cancellation by noise scaling. *Phys. Rev. A*, 104:052607, 2021. doi: 10.1103/PhysRevA.104.052607.

McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R., and Neven, H. Barren plateaus in quantum neural network training landscapes. *Nat. Commun.*, 9(1):1–6, 2018. doi: 10.1038/s41467-018-07090-4.

Meyer, N., Ufrecht, C., Periyasamy, M., Scherer, D. D., Plinge, A., and Mutschler, C. A Survey on Quantum Reinforcement Learning. *arXiv:2211.03464*, 2022. doi: 10.48550/arXiv.2211.03464.

Meyer, N., Scherer, D. D., Plinge, A., Mutschler, C., and Hartmann, M. J. Quantum Natural Policy Gradients: Towards Sample-Efficient Reinforcement Learning. *arXiv:2304.13571*, 2023. doi: 10.48550/arXiv.2304.13571.

Mitarai, K., Negoro, M., Kitagawa, M., and Fujii, K. Quantum circuit learning. *Phys. Rev. A*, 98(3):032309, 2018. doi: 10.1103/PhysRevA.98.032309.

Moll, N., Barkoutsos, P., Bishop, L. S., Chow, J. M., Cross, A., Egger, D. J., Filipp, S., Fuhrer, A., Gambetta, J. M., Ganzhorn, M., et al. Quantum optimization using variational algorithms on near-term quantum devices. *Quantum Sci. Technol.*, 3(3):030503, 2018. doi: 10.1088/2058-9565/aab822.

Nation, P. D., Kang, H., Sundaresan, N., and Gambetta, J. M. Scalable mitigation of measurement errors on quantum computers. *PRX Quantum*, 2(4):040326, 2021. doi: 10.1103/PRXQuantum.2.040326.

Nielsen, M. A. *Neural Networks and Deep Learning*. Determination Press, 2015. URL https://books.google.de/books?id=STDBswEACAAJ.

Nielsen, M. A. and Chuang, I. L. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010. doi: 10.1017/CBO9780511976667.

Pérez-Salinas, A., Cervera-Lierta, A., Gil-Fuster, E., and Latorre, J. I. Data re-uploading for a universal quantum classifier. *Quantum*, 4:226, 2020. doi: 10.22331/q-2020-02-06-226.

Periyasamy, M., Meyer, N., Ufrecht, C., Scherer, D. D., Plinge, A., and Mutschler, C. Incremental Data-Uploading for Full-Quantum Classification. In *IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 31–37, 2022. doi: 10.1109/QCE53715.2022.00021.

Poggio, T., Banburski, A., and Liao, Q. Theoretical issues in deep networks. *Proceedings of the National Academy of Sciences*, 117(48):30039–30045, 2020. doi: 10.1073/pnas.1907369117.

Preskill, J. Quantum Computing in the NISQ era and beyond. *Quantum*, 2:79, 2018. doi: 10.22331/q-2018-08-06-79.

Qiskit contributers. Qiskit: An Open-source Framework for Quantum Computing. https://quantum-computing.ibm.com/, 2023.

Reddi, S. J., Kale, S., and Kumar, S. On the Convergence of Adam and Beyond. *Proc. of the Int. Conf. on Learning Representations (ICLR)*, 2018. doi: 10.48550/arXiv.1904.09237.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986. doi: 10.1038/323533a0.

Saggio, V., Asenbeck, B. E., Hamann, A., Strömberg, T., Schiansky, P., Dunjko, V., Friis, N., Harris, N. C., Hochberg, M., Englund, D., et al. Experimental quantum speed-up in reinforcement learning agents. *Nature*, 591(7849):229–233, 2021. doi: 10.1038/s41586-021-03242-7.

Schuld, M. Supervised quantum machine learning models are kernel methods. *arXiv:2101.11020*, 2021. doi: 10.48550/arXiv.2101.11020.

Schuld, M. and Petruccione, F. *Supervised Learning with Quantum Computers*. Springer, 2018. doi: 10.1007/978-3-319-96424-9.

Schuld, M., Bergholm, V., Gogolin, C., Izaac, J., and Killoran, N. Evaluating analytic gradients on quantum hardware. *Phys. Rev. A*, 99(3):032331, 2019. doi: 10.1103/PhysRevA.99.032331.

Schuld, M., Sweke, R., and Meyer, J. J. Effect of data encoding on the expressive power of variational quantum-machine-learning models. *Phys. Rev. A*, 103(3):032430, 2021. doi: 10.1103/PhysRevA.103.032430.

Sequeira, A., Santos, L. P., and Barbosa, L. S. Variational Quantum Policy Gradients with an Application to Quantum Control. *arXiv:2203.10591*, 2022. doi: 10.48550/arXiv.2203.10591.

Sharma, K., Khatri, S., Cerezo, M., and Coles, P. J. Noise resilience of variational quantum compiling. *New J. Phys.*, 22(4):043006, 2020. doi: 10.1088/1367-2630/ab784c.

Sim, S., Johnson, P. D., and Aspuru-Guzik, A. Expressibility and Entangling Capability of Parameterized Quantum Circuits for Hybrid Quantum-Classical Algorithms. *Adv. Quantum Technol.*, 2(12):1900070, 2019. doi: 10.1002/qute.201900070.

Spall, J. C. An Overview of the Simultaneous Perturbation Method for Efficient Optimization. *Johns Hopkins APL Tech. Dig.*, 19(4):482–492, 1998. URL https://www.jhuapl.edu/SPSA/.

Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. MIT press, 2018. URL http://incompleteideas.net/book/the-book.html.

Sutton, R. S., McAllester, D., Singh, S., and Mansour, Y. Policy Gradient Methods for Reinforcement Learning with Function Approximation. In *Advances in Neural Information Processing Systems*, volume 12, 1999. URL https://papers.nips.cc/paper/1999.

Sweke, R., Seifert, J.-P., Hangleiter, D., and Eisert, J. On the Quantum versus Classical Learnability of Discrete Distributions. *Quantum*, 5:417, 2021. doi: 10.22331/q-2021-03-23-417.

Wiedmann, M., Hölle, M., Periyasamy, M., Meyer, N., Ufrecht, C., Scherer, D. D., Plinge, A., and Mutschler, C. An Empirical Comparison of Optimizers for Quantum Machine Learning with Spsa-based Gradients. *arXiv:2305.00224*, 2023. doi: 10.48550/arXiv.2305.00224.

Wilkinson, S. A. and Hartmann, M. J. Evaluating the performance of sigmoid quantum perceptrons in quantum neural networks. *arXiv:2208.06198*, 2022. doi: 10.48550/arXiv.2208.06198.

Wu, S., Jin, S., Wen, D., and Wang, X. Quantum reinforcement learning in continuous action space. *arXiv:2012.10711*, 2020. doi: 10.48550/arXiv.2012.10711.

Wu, Y., Bao, W.-S., Cao, S., Chen, F., Chen, M.-C., Chen, X., Chung, T.-H., Deng, H., Du, Y., Fan, D., et al. Strong quantum computational advantage using a superconducting quantum processor. *Phys. Rev. Lett.*, 127(18):180501, 2021. doi: 10.1103/PhysRevLett.127.180501.

Zhong, H.-S., Wang, H., Deng, Y.-H., Chen, M.-C., Peng, L.-C., Luo, Y.-H., Qin, J., Wu, D., Ding, X., Hu, Y., et al. Quantum computational advantage using photons. *Science*, 370(6523):1460–1463, 2020. doi: 10.1126/science.abe8770.

A. Caveats of the SOFTMAX-VQC Policy

In Section 3.2 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., 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 β in the SOFTMAX-VQC policy equation

$$\pi_{\boldsymbol{\lambda},\boldsymbol{\theta}}(a \mid \boldsymbol{s}) = \frac{e^{\beta \langle O_a \rangle_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}}}}{\sum_{a' \in \mathcal{A}} e^{\beta \langle O_{a'} \rangle_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}}}}, \tag{19}$$

where $\langle O_a \rangle_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}} := \langle \psi_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}} | O_a | \psi_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}} \rangle$, 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 Equation (19) 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. (Jerbi et al., 2021) 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 $\boldsymbol{s} \in \mathbb{R}^n$, 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 $|\psi_{\boldsymbol{s},\boldsymbol{\theta},\boldsymbol{\lambda}}\rangle = U_{\boldsymbol{s},\boldsymbol{\theta},\boldsymbol{\lambda}} |0\rangle^{\otimes n}$, we define its associated RESTRICTED-SOFTMAX-VQC policy as:

$$\pi_{\Theta}(a \mid \boldsymbol{s}) = \frac{e^{\beta w_a \langle O \rangle_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}}}}{\sum_{a'} e^{\beta w_{a'} \langle O \rangle_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}}}} \tag{20}$$

where $\langle O \rangle_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}} = \langle \psi_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}} \mid O \mid \Psi_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}} \rangle$ 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. $\Theta = (\boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{w})$ constitute all the trainable parameters of this policy.

For completeness, we demonstrate in Appendix 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 Appendix A.2. In Appendix A.3, we extract some implications for the original SOFTMAX-VQC policy from the findings.

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

It follows directly from Equation (20), that the action selection procedure requires the estimation of only a single expectation value, namely $\langle O \rangle_{\boldsymbol{s},\boldsymbol{\lambda},\boldsymbol{\theta}}$. It is not directly obvious, if this reduction also translates to the gradient estimation, more concretely to the log-policy gradient required for Equation (1). Following Definition A.1, the gradients w.r.t. $\boldsymbol{\theta}$ of a RESTRICTED-SOFTMAX-VQC simplify to

$$\begin{aligned}
& \nabla_{\boldsymbol{\theta}} \ln \pi_{\Theta}(a \mid \boldsymbol{s}) && (21)\\
= \; & \beta \cdot \left(\nabla_{\boldsymbol{\theta}} w_a \langle O \rangle_{\boldsymbol{s},\boldsymbol{\theta},\boldsymbol{\lambda}} - \sum_{a'} \pi_{\Theta}(a' \mid \boldsymbol{s}) \cdot \nabla_{\boldsymbol{\theta}} w_{a'} \langle O \rangle_{\boldsymbol{s},\boldsymbol{\theta},\boldsymbol{\lambda}} \right) && (22)\\
= \; & \beta \cdot \nabla_{\boldsymbol{\theta}} \langle O \rangle_{\boldsymbol{s},\boldsymbol{\theta},\boldsymbol{\lambda}} \left(w_a - \sum_{a'} \pi_{\Theta}(a' \mid \boldsymbol{s}) \cdot w_{a'} \right). && (23)
\end{aligned}$$

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

$$\begin{aligned} & \nabla_{w_x} \ln \pi_\Theta(a \mid \boldsymbol{s}) && (24) \\ = \; & \beta \cdot \left(\nabla_{w_x} w_a \langle O \rangle_{\boldsymbol{s},\boldsymbol{\theta},\boldsymbol{\lambda}} - \sum_{a'} \pi_\Theta(a' \mid \boldsymbol{s}) \cdot \nabla_{w_x} w_{a'} \langle O \rangle_{\boldsymbol{s},\boldsymbol{\theta},\boldsymbol{\lambda}} \right) && (25) \\ = \; & \beta \cdot \langle O \rangle_{\boldsymbol{s},\boldsymbol{\theta},\boldsymbol{\lambda}} \left(\delta_{a=x} - \pi_\Theta(x \mid \boldsymbol{s}) \right), && (26) \end{aligned}$$

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.

A.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 π_*, each of the state sets must be equally likely to observe. With the notion of expected fraction of time spend in state $\boldsymbol{s}$ as $\mu(\boldsymbol{s})$ from Sutton & Barto, this is stated as $\sum_{\boldsymbol{s} \in \mathcal{S}_i} \mu(\boldsymbol{s}) = \frac{1}{|\mathcal{A}|}$. Let now $\mathcal{S}_i$ (with $i \in \{0,\ 1,\ ...,\ |\mathcal{A}| - 1\}$) be an arbitrary state set and $\boldsymbol{s}$ an arbitrary state from this set. It must hold, that $\pi_*(a_i \mid \boldsymbol{s}) = 1$, and consequently $\pi_*(a_j \mid \boldsymbol{s}) = 0$ 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 π.

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 $\mathcal{S}_2 = \{2, 3\}$, 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 Definition A.2.

Lemma A.3. *Let π be any `RESTRICTED-SOFTMAX`-VQC policy, with w.l.o.g. $\langle O \rangle_{\boldsymbol{s}} \in [-1, 1]$, for all $\boldsymbol{s} \in \mathcal{S}$. Given a uniform environment $\mathcal{E}_\mathcal{A}$, the performance of the model is upper bounded by*

$$ACC_\pi(\mathcal{E}_\mathcal{A}) \leq \frac{2}{|\mathcal{A}|} \sum_{k=1}^{|\mathcal{A}|/2} \frac{1}{k}. \tag{27}$$

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

$$w_0 \geq w_1 \geq \cdots \geq w_{n-2} \geq w_{n-1}. \tag{28}$$

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

$$\begin{aligned} & \frac{e^{w_k \langle O \rangle_{\boldsymbol{s}}}}{\sum_{a'} e^{w_{a'} \langle O \rangle_{\boldsymbol{s}}}} && > && \frac{1}{k} && && (29) \\ \Leftrightarrow \quad & (k-1) \cdot e^{w_k \langle O \rangle_{\boldsymbol{s}}} && > && \sum_{a' \neq k} e^{w_{a'} \langle O \rangle_{\boldsymbol{s}}} && && (30) \\ \Leftrightarrow \quad & k - 1 && > && \underbrace{\frac{e^{w_0 \langle O \rangle_{\boldsymbol{s}}}}{e^{w_k \langle O \rangle_{\boldsymbol{s}}}}}_{\geq 1} + \cdots + \underbrace{\frac{e^{w_{n-1} \langle O \rangle_{\boldsymbol{s}}}}{e^{w_k \langle O \rangle_{\boldsymbol{s}}}}}_{\geq 1} && \geq \quad k - 1 \quad \lightning && (31) \end{aligned}$$

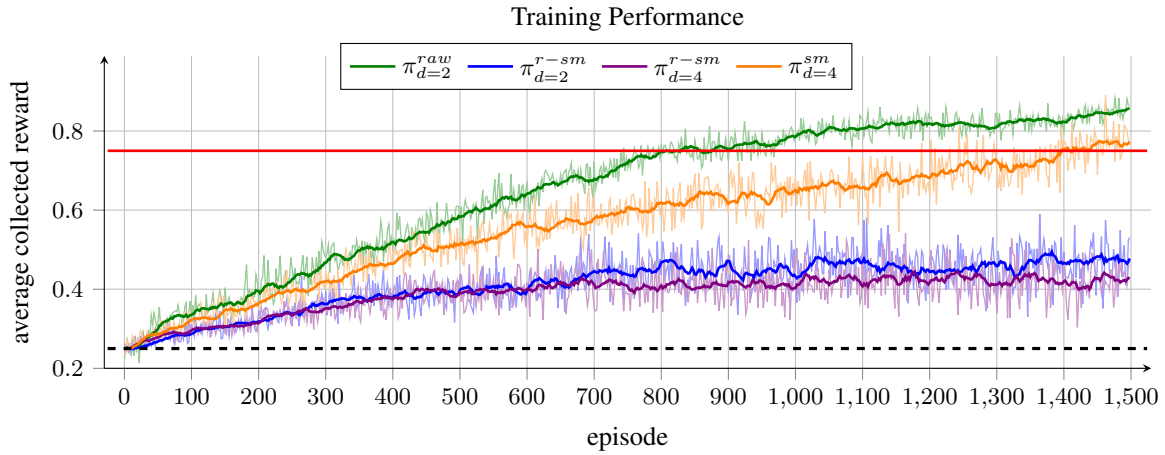


Figure 6. Experiments on a `ContextualBandits` environment (which is uniform following Definition A.2) demonstrate the implications of Lemma A.3. The reward structure is defined in a way, such that the expected reward is equivalent to the introduced notion of accuracy $ACC_\pi(\mathcal{E}_\mathcal{A})$. A `RESTRICTED-SOFTMAX`-VQC π^{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 π^{raw} and `SOFTMAX`-VQC π^{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 Section 4. All results are averaged over 50 independent experiments.

The last step uses Equation (28) combined with the monotonicity of the exponential function. As $\mathcal{E}_\mathcal{A}$ is an uniform environment, for the described policy it holds $ACC_\pi\left(\mathcal{E}_\mathcal{A}\right) \leq \frac{1}{|\mathcal{A}|}\sum_{k=1}^{|\mathcal{A}|}\frac{1}{k}$. An improvement of this bound can be achieved by allowing $\langle O\rangle_{\boldsymbol{s}} \in [-1, 1]$. Multiplying with a negative value inverts the inequality chain from Equation (28) to $-w_0 \leq -w_1 \leq \cdots \leq -w_{n-2} \leq -w_{n-1}$, which introduces the missing factor of 0.5 into Equation (27). 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 $\langle O\rangle_{\boldsymbol{s}} = 0$ does not lead to any improvement, as all actions will be selected with equal probability. □

This upper bound on performance makes the `RESTRICTED-SOFTMAX`-VQC unsuited for uniform environments with large action spaces. In fact, for $|\mathcal{A}| \to \infty$ the accuracy converges to 0. Already for 4 actions, the accuracy is bounded by $\frac{2}{4}\left(\frac{1}{1} + \frac{1}{2}\right) = \frac{3}{4}$, experimental results on the `ContextualBandits` environment described above are depicted in Figure 6. We expect this result to hold in a weakened form for more general environments.

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

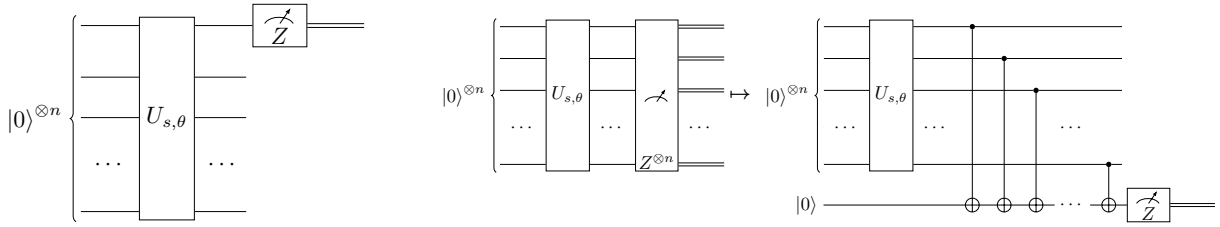
Directly following from Lemma A.3, one can make also a statement about the non-restricted `SOFTMAX`-VQC policy:

Corollary A.4. *Let π be any `SOFTMAX`-VQC policy and $\mathcal{E}_\mathcal{A}$ a uniform environment. In order to not impose any constraints following Lemma A.3, 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 Corollary A.4, 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.5. *In order to not put any performance constraints on the `SOFTMAX`-VQC policy by construction, it has to incorporate $\mathcal{O}\left(|\mathcal{A}|\right)$* different *observables O_a, i.e. the number has to scale linearly in the number of actions.*

Unfortunately, a naive interpretation of Corollary A.5 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 Section 5 provide adequate justification for the focus on this formulation.



(a) Measures only a 1-qubit Pauli-Z observable on the first qubit (associated with the *local* policy in Equation (41)).

(b) Measures a tensored Pauli-Z observable on all qubits (associated with the *global* policy in Equation (42)). The depicted construction of CX gates and a 1-qubit Pauli-Z measurement on an ancilla qubit defines a equivalent policy.

Figure 7. Two types of observables, that give rise to different policy formulations.

B. Action Decoding with Local and Global Observables

Continuing with the considerations from Section 4.1, for an environment with 2 actions, we have to define $\mathcal{V}_{a=0}$ and $\mathcal{V}_{a=1}$. Following Jerbi et al. for a 3-qubit system yields:

$$\mathcal{V}^{\text{loc}}_{a=0} = \{|000\rangle, |001\rangle, |010\rangle, |011\rangle\} \tag{32}$$
$$\mathcal{V}^{\text{loc}}_{a=1} = \{|100\rangle, |101\rangle, |110\rangle, |111\rangle\} \tag{33}$$

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

$$\mathcal{V}^{\text{loc}}_{a} = \{|b_{n-1} b_{n-2} \cdots b_1 b_0\rangle \mid a = b_{n-1}\}. \tag{34}$$

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

$$\mathcal{V}^{\text{glob}}_{a=0} = \{|000\rangle, |011\rangle, |101\rangle, |110\rangle\} \tag{35}$$
$$\mathcal{V}^{\text{glob}}_{a=1} = \{|001\rangle, |010\rangle, |100\rangle, |111\rangle\}, \tag{36}$$

and more generally for n qubits:

$$\mathcal{V}^{\text{glob}}_{a} = \left\{ |b_{n-1} b_{n-2} \cdots b_1 b_0\rangle \mid a = \bigoplus_{i=0}^{n-1} b_i \right\}. \tag{37}$$

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 Equations (34) and (37), it is possible to instead model the policy in terms of expectation values of observables:

$$\pi_\Theta(a|\boldsymbol{s}) = \sum_{v \in \mathcal{V}_a} \langle \psi_{\boldsymbol{s},\Theta} \mid v \rangle \langle v \mid \psi_{\boldsymbol{s},\Theta} \rangle \tag{38}$$
$$= \frac{\sum_{v \in \mathcal{V}_a} \langle \psi_{\boldsymbol{s},\Theta} \mid v \rangle \langle v \mid \psi_{\boldsymbol{s},\Theta} \rangle - \left(1 - \sum_{v \in \mathcal{V}_a} \langle \psi_{\boldsymbol{s},\Theta} \mid v \rangle \langle v \mid \psi_{\boldsymbol{s},\Theta} \rangle\right) + 1}{2} \tag{39}$$
$$= \frac{\sum_{v \in \mathcal{V}_a} \langle \psi_{\boldsymbol{s},\Theta} \mid v \rangle \langle v \mid \psi_{\boldsymbol{s},\Theta} \rangle - \sum_{v \in \mathcal{V}_{\tilde{a}}} \langle \psi_{\boldsymbol{s},\Theta} \mid v \rangle \langle v \mid \psi_{\boldsymbol{s},\Theta} \rangle + 1}{2}, \tag{40}$$

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 $\mathcal{V}^{\text{loc}}_{a=0}$, as defined in Equation (34). Vice versa, in all other cases, the measurement outputs a value of -1. This simplifies the above equation to

$$\pi^{\text{loc}}_\Theta(a|\boldsymbol{s}) = \frac{(-1)^a \cdot \left\langle \psi_{\boldsymbol{s},\Theta} \mid Z \otimes I^{\otimes n-1} \mid \psi_{\boldsymbol{s},\Theta} \right\rangle + 1}{2}. \tag{41}$$

Returning to the projector formalism from Definition 3.1, this can alternatively also be expressed as $\pi_\Theta^{\text{loc}}(a|\boldsymbol{s}) = \left\langle \psi_{\boldsymbol{s},\Theta} \left| \sum_{\boldsymbol{b}\in\{0,1\}^{n-1}} |a\rangle |\boldsymbol{b}\rangle \langle a| \langle \boldsymbol{b}| \right| \psi_{\boldsymbol{s},\Theta} \right\rangle$, 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

$$\pi_\Theta^{\text{glob}}(a \mid \boldsymbol{s}) = \frac{(-1)^a \cdot \langle \psi_{\boldsymbol{s},\Theta} \mid Z^{\otimes n} \mid \psi_{\boldsymbol{s},\Theta}\rangle + 1}{2}. \tag{42}$$

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

As also visualized in Figure 7, 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 Figure 7 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

$$\pi_\Theta^{\text{q-loc}}(a \mid \boldsymbol{s}) = \frac{(-1)^a \cdot \langle \psi_{\boldsymbol{s},\Theta} \mid Z^{\otimes q} \otimes I^{\otimes n-q} \mid \psi_{\boldsymbol{s},\Theta}\rangle + 1}{2}, \tag{43}$$

where $\pi_\Theta^{\text{n-loc}}$ is equivalent to π_Θ^{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 Section 5 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 Equations (41) to (43) 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 Equation (42), 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 Figure 7. This ancilla qubit is initialized to $|0\rangle$, and after the evolution of the system with $U_{\boldsymbol{s},\Theta}$, it interacts via a CX-gate with each of the n original qubits.

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 Section 4.2.1:

$$\mathcal{C}_{a=0} = \{0000,\ 0010,\ 0100,\ 0110\} \tag{44}$$
$$\mathcal{C}_{a=1} = \{0001,\ 0011,\ 0101,\ 0111\} \tag{45}$$
$$\mathcal{C}_{a=2} = \{1000,\ 1010,\ 1101,\ 1111\} \tag{46}$$
$$\mathcal{C}_{a=3} = \{1001,\ 1011,\ 1100,\ 1110\} \tag{47}$$

As the system is really small, it is straightforward to determine the extracted information following Definition 4.1 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.

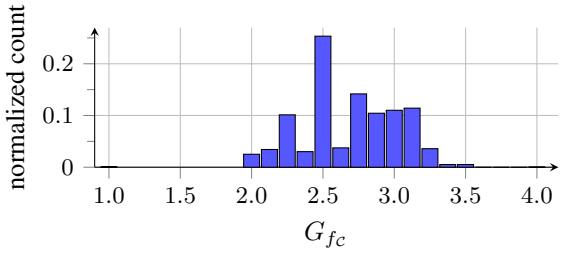
With this work done, it is straightforward to compute the associated globality measure following Equation (12) as

$$G_{f_c} = \frac{1}{2^4} \sum_{\boldsymbol{b}\in\{0,1\}^4} EI_{f_c}(\boldsymbol{b}) \tag{48}$$
$$= \frac{8 \cdot 2 + 8 \cdot 3}{16} = 2.5. \tag{49}$$

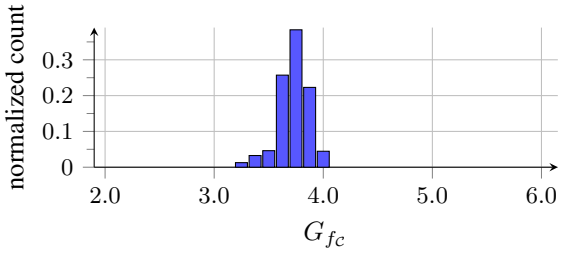
The result reads itself as that on average 2.5 bit of information is necessary to make an unambiguous action assignment. This is obviously above the minimum value of $G_{f_{n=4}}^{\min} = \log_2(4) = 2$, but well below the optimum of $G_{f_{n=4}}^{\max} = n = 4$. In Appendix D.3 we demonstrated how a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

Table 2. Extracted information of the post-processing function $f_{\mathcal{C}}$ for the partitioning $\mathcal{C}$ given in Equations (44) to (47) 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.

bitstring $\boldsymbol{b}$	$EI_{f_{\mathcal{C}}}(\boldsymbol{b})$	containing partition	bitstring $\boldsymbol{b}$	$EI_{f_{\mathcal{C}}}(\boldsymbol{b})$	containing partition
$\boxed{0}\,0\,0\,\boxed{0}$	2	$\mathcal{C}_{a=0}$	$\boxed{1}\,\boxed{0}\,0\,\boxed{0}$	3	$\mathcal{C}_{a=2}$
$\boxed{0}\,0\,0\,\boxed{1}$	2	$\mathcal{C}_{a=1}$	$\boxed{1}\,\boxed{0}\,0\,\boxed{1}$	3	$\mathcal{C}_{a=3}$
$\boxed{0}\,0\,\boxed{1}\,0$	2	$\mathcal{C}_{a=0}$	$\boxed{1}\,\boxed{0}\,\boxed{1}\,0$	3	$\mathcal{C}_{a=2}$
$\boxed{0}\,0\,\boxed{1}\,\boxed{1}$	2	$\mathcal{C}_{a=1}$	$\boxed{1}\,\boxed{0}\,\boxed{1}\,\boxed{1}$	3	$\mathcal{C}_{a=3}$
$\boxed{0}\,\boxed{1}\,0\,0$	2	$\mathcal{C}_{a=0}$	$\boxed{1}\,\boxed{1}\,0\,0$	3	$\mathcal{C}_{a=3}$
$\boxed{0}\,\boxed{1}\,0\,\boxed{1}$	2	$\mathcal{C}_{a=1}$	$\boxed{1}\,\boxed{1}\,0\,\boxed{1}$	3	$\mathcal{C}_{a=2}$
$\boxed{0}\,\boxed{1}\,\boxed{1}\,0$	2	$\mathcal{C}_{a=0}$	$\boxed{1}\,\boxed{1}\,\boxed{1}\,0$	3	$\mathcal{C}_{a=3}$
$\boxed{0}\,\boxed{1}\,\boxed{1}\,\boxed{1}$	2	$\mathcal{C}_{a=1}$	$\boxed{1}\,\boxed{1}\,\boxed{1}\,\boxed{1}$	3	$\mathcal{C}_{a=2}$



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



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

Figure 8. Histogram of globality values over all possible partitionings.

D. Supplementary Material on Construction of an Optimal Partitioning

As discussed throughout Section 4.2, 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 Equation (12). However, this property is highly desirable, as it strongly correlates with RL performance, as demonstrated in Section 5.1.

D.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 $G_{f_{\mathcal{C}}} = n$, as shown in Figure 8.

D.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 Lemma 4.2 below:

Lemma D.1. *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 Equations (13) to (15),*

one can define

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

where K is the number of shots, $\boldsymbol{b}^{(k)}$ is the bitstring observed in the k-th experiment, and δ 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 Equations (14) and (15) is trivial, as it corresponds to the previous considerations from Equations (37) and (42). The induction step $m \to m+1$ needs to consider the two sets, into which $\mathcal{C}_{[a]_2}^{(m)}$ gets decomposed by inversely applying Equation (13):

$$\mathcal{C}_{a_m \cdots a_1 0 a_0}^{(m+1)} = \left\{ \boldsymbol{b} = b_{n-1} b_{n-2} \cdots b_1 b_0 \mid \boxed{\bigoplus_{i=m+1}^{n-1} b_i = a_0} \wedge \boldsymbol{b} \in \mathcal{C}^{(m)}_{\underbrace{a_m \cdots a_1 \, (0 \oplus a_0)}_{[a]_2}} \right\} \tag{50}$$

$$\mathcal{C}_{a_m \cdots a_1 1 \tilde{a}_0}^{(m+1)} = \left\{ \boldsymbol{b} = b_{n-1} b_{n-2} \cdots b_1 b_0 \mid \boxed{\bigoplus_{i=m+1}^{n-1} b_i = \tilde{a}_0} \wedge \boldsymbol{b} \in \mathcal{C}^{(m)}_{\underbrace{a_m \cdots a_1 \, (1 \oplus \tilde{a}_0)}_{[a]_2}} \right\}, \tag{51}$$

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 $G_{f_{\mathcal{C}}} = n$. □

D.3. Example of Optimal Partitioning

To get some intuition for the construction arising from Lemma 4.2, 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 $m = \log_2(4) - 1 = 1$.

$$\mathcal{C}^{(1)}_{[0]_2=00} = \left\{ \boldsymbol{b} \mid \boxed{b_3 \oplus b_2 \oplus b_1 = 0} \wedge \boldsymbol{b} \in \mathcal{C}^{(0)}_{0 \oplus 0 = [0]_2} \right\} = \left\{ \boxed{000}\,0 \,, \boxed{011}\,0 \,, \boxed{101}\,0 \,, \boxed{110}\,0 \right\} \tag{52}$$

$$\mathcal{C}^{(1)}_{[3]_2=11} = \left\{ \boldsymbol{b} \mid \boxed{b_3 \oplus b_2 \oplus b_1 = 1} \wedge \boldsymbol{b} \in \mathcal{C}^{(0)}_{1 \oplus 1 = [0]_2} \right\} = \left\{ \boxed{001}\,1 \,, \boxed{010}\,1 \,, \boxed{100}\,1 \,, \boxed{111}\,1 \right\} \tag{53}$$

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

$$\mathcal{C}^{(0)}_{[0]_2} = \{ \boldsymbol{b} \mid b_3 \oplus b_2 \oplus b_1 \oplus b_0 = 0 \} \tag{54}$$

The construction for the remaining two partitions works totally equivalent:

$$\mathcal{C}^{(1)}_{[1]_2=01} = \left\{ \boldsymbol{b} \mid \boxed{b_3 \oplus b_2 \oplus b_1 = 1} \wedge \boldsymbol{b} \in \mathcal{C}^{(0)}_{0 \oplus 1 = [1]_2} \right\} = \left\{ \boxed{001}\,0 \,, \boxed{010}\,0 \,, \boxed{100}\,0 \,, \boxed{111}\,0 \right\} \tag{55}$$

$$\mathcal{C}^{(1)}_{[2]_2=10} = \left\{ \boldsymbol{b} \mid \boxed{b_3 \oplus b_2 \oplus b_1 = 0} \wedge \boldsymbol{b} \in \mathcal{C}^{(0)}_{1 \oplus 0 = [1]_2} \right\} = \left\{ \boxed{000}\,1 \,, \boxed{011}\,1 \,, \boxed{101}\,1 \,, \boxed{110}\,1 \right\} \tag{56}$$

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

$$\mathcal{C}^{(0)}_{[1]_2} = \{ \boldsymbol{b} \mid b_3 \oplus b_2 \oplus b_1 \oplus b_0 = 1 \} \tag{57}$$

We are guaranteed by Lemma 4.2 that $G_{f_\mathcal{C}} = 4$, 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 $\mathcal{C}^{(2)}_{[5]_2=101} = \left\{ \boldsymbol{b} \mid b_3 \oplus b_2 = 1 \wedge \boldsymbol{b} \in \mathcal{C}^{(1)}_{1(0\oplus 1)=[3]_2} \right\} = \{0101, 1001\}$. Equation (18) allows to determine the class-association of an arbitrary bitstring without explicitly constructing the partitioned sets. For an setup with 8 actions, i.e. $m = \log_2 8 - 1 = 2$ we get

$$f_\mathcal{C}(\boldsymbol{b}) = [10\,(1 \oplus 0)]_{10} = 5, \tag{58}$$

which correctly identifies 1001 to be an element of partition $\mathcal{C}^{(2)}_{[5]_2}$.

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.. 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 $p_{\pi_\Theta}(\boldsymbol{s}, a) = \pi_\Theta(a \mid \boldsymbol{s})p_{\pi_\Theta}(\boldsymbol{s})$, where the prior state distribution $p_{\pi_\Theta} : \mathcal{S} \to [0, 1]$ 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 (Abbas et al., 2021) 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 Φ. From the full parameter space Φ each individual parameter set Θ 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

$$\tilde{F}_k(\Theta) = \frac{1}{k} \sum_{i=1}^{k} \left[\nabla_\Theta \ln p_{\pi_\Theta}(s_i, a_i) \nabla_\Theta \ln p_{\pi_\Theta}(s_i, a_i)^t \right]. \tag{59}$$

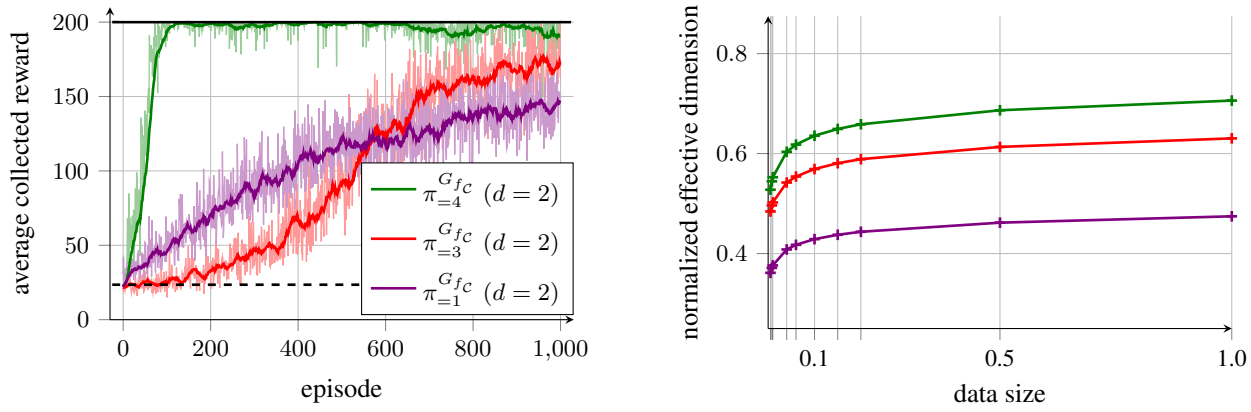
An alternate formulation from Sequeira et al. drops the dependence on the prior state distribution. More concretely, this reduces Equation (59) to $\tilde{F}_k(\Theta) = \frac{1}{k} \sum_{i=1}^{k} \left[\nabla_\Theta \ln \pi_\Theta\left(a_i \mid s_i\right) \nabla_\Theta \ln \pi_\Theta\left(a_i \mid s_i\right)^t \right]$. However, we assume that keeping the potentially inaccurate prior state information still should be beneficial.

E.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* (Abbas et al., 2021) 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

$$ed_n(\mathcal{M}_\Phi) := 2 \frac{\ln\left(\frac{1}{V_\Phi} \int_\Phi \sqrt{\det\left(I_{|\Theta|} + \frac{n}{2\pi \ln n} \hat{F}(\Theta) \right)} d\Theta \right)}{\ln\left(\frac{n}{2\pi \ln n} \right)}. \tag{60}$$

The FIM $\hat{F}(\Theta)$ is normalized, such that $\frac{1}{V_\Phi} \int_\Phi \operatorname{tr}\left(\hat{F}(\Theta) \right) d\Theta = |\Theta|$ 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.



(a) RL training performance

(b) Normalized Effective dimension

Figure 9. RL training performance and associated effective dimension on the `CartPole-v0` environment. The same setup and policy formulations as in Figure 3 are used, only the depth of the underlying VQCs is increased to $d = 2$.

E.2. Trainability of the VQC-Model

The spectrum of the FIM, i.e. its eigenvalue distribution, provides insights into the trainability of a model (Abbas et al., 2021). 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 Equation (59).

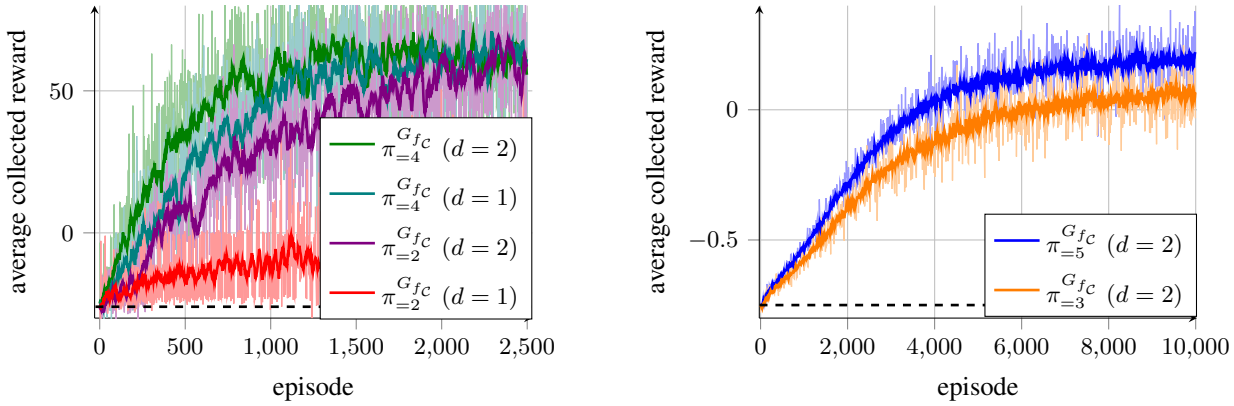
F. Supplementary Experiments and Conventions

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) (Franz et al., 2022). To start with, all experiments on the `CartPole-v0` environment use a learning rate of $\alpha_{\boldsymbol{\theta}} = 0.01$ for the variational and $\alpha_{\boldsymbol{\lambda}} = 0.1$ 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 (Kingma & Ba, 2015), modified with the *AMSGrad* adjustment (Reddi et al., 2018). 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 Figure 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 Section 5, 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.

F.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 Figure 9. Compared to the results in Figure 3, the convergence speed of the policies $\pi_{=3}^{G_{fc}}$ and $\pi_{=1}^{G_{fc}}$ 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



(a) `FrozenLake` environment with 16 discrete states and 4 actions.

(b) `ContextualBandits` environment with 32 discrete states and 8 actions.

Figure 10. RL training performance for different environments. The underlying global post-processing functions are determined as described in Section 4.2.2. These results are averaged over 100 independent experiments;

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 $\pi_{=4}^{G_{fc}}$ 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 Figure 9. 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.

F.2. Extension to Other Environments

To really make use of the construction of global post-processing functions for larger action spaces proposed by Lemma 4.2, we now take a look at two additional environments. The first one is the gridworld environment `FrozenLake` (Brockman et al., 2016), 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 Figure 10. 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 Figure 10. 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.

Table 3. 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 $\boldsymbol{s}$. Unlike in Figure 4, the states elements are sampled uniformly at random from $[-\pi, \pi)$ to allow for statements abstracted from a concrete RL environment.

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

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 range of different setups. We keep things as general as possible by sampling state values uniformly at random from $(-\pi, \pi]^{|\boldsymbol{s}|}$. 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 Table 3. 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 $G_\mathcal{C} = 1$ are quite degenerated. As one would expect, the post-processing functions with $G_\mathcal{C} = 3$ 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.

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 (McClean et al., 2018). Abbas et al. relates this to the Fisher information spectrum, i.e. the model is vulnerable to barren plateaus, iff $\mathrm{tr}\left(\mathbb{E}_\Theta\left[F(\Theta)\right]\right)$ decreases exponentially with increasing system size (McClean et al., 2018). Following Table 3, no setup shows a progressive convergence of eigenvalues to 0, although there are some quantitative differences. 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 small-scale models with global post-processing functions. Similar observations have been made in other fields of QML (Abbas et al., 2021; Kashif & Al-Kuwari, 2023). However, the interpretation of the global post-processing function as global measurement potentially makes barren plateaus inevitable for increasing system size as shown in Cerezo et al. – although the validity for large action spaces is not immediate. For a larger qubit count and circuit depth the proposed post-processing technique also allows adjusting the globality. This can be used to find a good balance between the empirical performance improvement demonstrated in this paper and reduced globality to prevent barren plateaus.

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 (Hochreiter, 1998). If the results presented in this section can be extended to larger quantum systems, the improvement in terms of trainability compared to classical models might point towards a possible quantum advantage.