Quantum Deep Equilibrium Models

Philipp Schleich*
Department of Computer Science
University of Toronto
Vector Institute
philipps@cs.toronto.edu

Marta Skreta*
Department of Computer Science
University of Toronto
Vector Institute
martaskreta@cs.toronto.edu

Lasse B. Kristensen
Department of Computer Science
University of Copenhagen

Rodrigo A. Vargas-Hernández
Department of Chemistry
& Chemical Biology
McMaster University, ON

Alán Aspuru-Guzik
Department of Computer Science
Department of Chemistry
University of Toronto
Vector Institute

Abstract

The feasibility of variational quantum algorithms, the most popular correspondent of neural networks on noisy, near-term quantum hardware, is highly impacted by the circuit depth of the involved parametrized quantum circuits (PQCs). Higher depth increases expressivity, but also results in a detrimental accumulation of errors. Furthermore, the number of parameters involved in the PQC significantly influences the performance through the necessary number of measurements to evaluate gradients, which scales linearly with the number of parameters. Motivated by this, we look at deep equilibrium models (DEQs), which mimic an infinite-depth, weight-tied network using a fraction of the memory by employing a root solver to find the fixed points of the network. In this work, we present Quantum Deep Equilibrium Models (QDEQs): a training paradigm that learns parameters of a quantum machine learning model given by a PQC using DEQs. To our knowledge, no work has yet explored the application of DEQs to QML models. We apply QDEQs to find the parameters of a quantum circuit in two settings: the first involves classifying MNIST-4 digits with 4 qubits; the second extends it to 10 classes of MNIST, FashionMNIST and CIFAR. We find that QDEQ is not only competitive with comparable existing baseline models, but also achieves higher performance than a network with 5 times more layers. This demonstrates that the QDEQ paradigm can be used to develop significantly more shallow quantum circuits for a given task, something which is essential for the utility of near-term quantum computers. Our code is available at https://github.com/martaskrt/qdeq.

*Equal contribution. Author order was sampled from a quantum computer.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-1004

1 Introduction

Quantum computing holds a lot of theoretical promise for transforming the computational landscape of a variety of applications. Machine learning is one of these applications (Biamonte et al., 2017), enabled by the capability of quantum computing to handle large amounts of data with significant quantum speedups and by the fact that dimensionality of the output is typically much smaller than the one of the input and during processing, a requirement to recover linear-algebra relevant speedups (Aaronson, 2015). Although there have been tremendous advances in hardware development in recent years, the time of full-stack, error corrected quantum computers which would enable such linear-algebra speedups is still not yet within tangible reach. Therefore, advances in near-term quantum algorithms for noisy devices with short coherence times remain one of the most promising avenues toward effectively harnessing quantum computing for real-world applications.

The arguably most studied class of algorithms in this line of research is the class of variational quantum algorithms (VQAs). While they originated as a means of physics-inspired learning without data (Peruzzo et al., 2014), they provide the most natural way to extend classical learning models to a hybrid quantum-classical setting. In VQAs, a parametrized quantum circuit (PQC) together with measurement of suitable observables on the quantum state after the circuit gives rise to a function class. Through the choice of a quantum circuit, observables, and loss function, this setup allows different problems to be tackled, with classical optimization of the loss function as the training step. Within this work, we use a VQA for classification tasks. Challenges within VQA training continue to be limited circuit depth, which reduces the the expressibility and trainability of the circuit, and disadvantageous loss landscapes, influenced by induced entanglement and the specific loss function (McClean et al., 2018; Fontana et al., 2024; Ragone et al., 2024). Further, while the parameter-shift rule allows us to evaluate gradients exactly using a simple finite-difference formula, there is a significant measurement overhead to determine the gradients. Thus, striving for training methods that enable the use of shallower circuits with less independent parameters at similar performance is paramount.

The main contribution of our work is the proposition of a class of quantum deep equilibrium models (QDEQ). To our knowledge, this is the first application of deep equilibrium models (DEQ) as first introduced in Bai et al. (2019) in the context of quantum computing; so far, previous work has explored implicit differentiation techniques for PQCs (Ahmed et al., 2022). Implicit and adjoint methods such as DEQs in machine learning stand out by their memory efficiency compared to their explicit counterparts. While this is still true for QDEQ, we see the main benefit in the reasons outlined in the paragraph above, i.e., the ability to use shallower circuits with fewer independent parameters to tackle a given problem. We will give some theoretical intuition why DEQ can be expected to perform well on a specific family of quantum model functions in Section 2.3, design a set of numerical experiments in Section 3 and obtain numerical results that confirm this hypothesis in Section 4.

2 Background and related works

2.1 Deep Equilibrium Models

Based on the fact that a neural network comprised of L layers is equivalent to an input-injected, weight-tied network of the same depth, Bai et al. (2019) introduced the concept of Deep Equilibrium Models. Assuming that the respective layer $f_\theta(\cdot)$ has a fixed point (Winston and Kolter, 2020), instead of explicitly repeating the weight-tied layer L times, we solve

$$f_\theta(\mathbf{z}^\star; \mathbf{x}) - \mathbf{z}^\star = g_\theta(\mathbf{z}^\star; \mathbf{x}) = 0 \tag{1}$$

using a black-box root finder such as Newton's or Broyden's method. The latter is comprised by Newton iterations where the Jacobian is approximated by a finite difference formula (Broyden, 1965). Finding this fixed point corresponds to evaluating an infinitely deep network, in the sense that $\lim_{L\to\infty} \mathbf{z}^{(L)} = \lim_{L\to\infty} \underbrace{(f_\theta \circ \cdots \circ f_\theta)}_{L \text{ times}}(\mathbf{z}^{(0)}) = \mathbf{z}^\star$.

When training this model with respect to a loss function ℓ, even though we use a root-finder to evaluate the model, we still need to be able to differentiate the procedure for training. Instead of explicitly differentiating through the L repetitions of the same layer (we call this later the DIRECT solver), Bai et al. (2019) make use of the implicit function theorem (Krantz and Parks, 2013) to

perform implicit differentiation in the following sense. During backpropagation, the gradient of the loss function can be expressed as:

$$\frac{\partial \ell}{\partial \theta} = -\frac{\partial \ell}{\partial \mathbf{z}^\star}\left(J_{g_\theta}^{-1}\,|_{\mathbf{z}^\star}\right)\frac{\partial f_\theta(\mathbf{z}^\star;\mathbf{x})}{\partial \theta}, \tag{2}$$

with $-\left(J_{g_\theta}^{-1}\,|_{\mathbf{z}^\star}\right) = (I - J_{f_\theta}\,|_{\mathbf{z}^\star})^{-1}$. While the last factor in Eq. (2) can be computed by standard automatic differentiation through the model function, determining $-\frac{\partial \ell}{\partial \mathbf{z}^\star}\left(J_{g_\theta}^{-1}\,|_{\mathbf{z}^\star}\right)$ either requires assembling the full Jacobian and inverting it (which is quite expensive) or solving an additional root-finding problem during the backward pass in form of solving for $\mathbf{q}$ in the equation

$$\left(J_{g_\theta}^{\mathrm{T}}\,|_{\mathbf{z}^\star}\right)\mathbf{q}^{\mathrm{T}} + \left(\frac{\partial \ell}{\partial \mathbf{z}^\star}\right)^{\mathrm{T}} = 0, \tag{3}$$

with gradients computed efficiently using the vector-Jacobian product.

The overall procedure of a DEQ model thus is to minimize a loss function ℓ given input $\mathbf{x}$, target y, and a hypothesis class in the following way: In the backward pass, we use the gradient expression in Eq. (2) and determine the first factor using a black-box root finder on Eq. (3). Using these gradients, the model is trained. The model is then evaluated on new input by using a root-finder to solve Eq. (1). A brief overview of this procedure is in Fig. 1.

It was found that pre-training the DEQ model using a shallow (2-layer), weight-tied DIRECT approach – that is, explicit application and training of $f \circ f$ – can considerably aid the overall convergence behaviour (Bai et al., 2019, 2020). This pretraining step is a cheap way to give the model reasonable weights as a starting point (called a warm-up), but the model is too shallow to achieve the best performance. In this work, we will utilize this warm-up strategy, then use implicit differentiation to optimize the loss and obtain better accuracy. We call this the IMPLICIT+WARMUP solver, and refer to training using only implicit differentiation as the IMPLICIT solver.

An important step to ensure stability of the root-finding algorithm is regularizing the Jacobian, as introduced in Bai et al. (2021). Further advances include a convenient inclusion to `pytorch` through Geng and Kolter (2023) and certifiable (Li et al., 2022) and robust (Chu et al., 2023) DEQ models as well as DEQ for diffusion models (Pokle et al., 2022).

2.2 Quantum Models and Variational Algorithms for Classification

Variational quantum algorithms rely on a PQC over Q qubits, which span the finite-dimensional Hilbert space $\mathbb{C}^{2^Q}$. Then, $\{|i\rangle\}_{i=0}^{2^Q-1} \sim \{|i_1\rangle \otimes |i_2\rangle \otimes \cdots |i_n\rangle\}_{i_1,\ldots,i_n \in \{0,1\}}$ is an orthonormal basis for this space and $\langle i|$ is the conjugate transpose to a vector $|i\rangle$. We denote the PQC, parametrized by p parameters, by the Q-qubit unitary operation $U(\theta) \in \mathbb{C}^{2^Q \times 2^Q}$. Then, one can define an objective $\langle\phi| U^\dagger(\theta) M U(\theta) |\phi\rangle$ that is to be variationally minimized, induced by a Hermitian matrix $M \in \mathbb{C}^{2^Q \times 2^Q}$ and an initial state $|\phi\rangle \in \mathbb{C}^{2^Q}$ (Bharti et al., 2022; Cerezo et al., 2021). Quantum machine learning models have already been the target of a fair amount of study, yielding insights into several properties, including generalization bounds (Caro et al., 2022), their training in general (Beer et al., 2020), and interpretability (Pira and Ferrie, 2024).

So far, we have not discussed how this framework is able to incorporate data into the training; here, this role will be filled by the initial state $|\phi\rangle$. Given classical data $\mathbf{x} \in \mathbb{R}^n$, there exists a unitary data encoding circuit $S_\mathbf{x}$ that maps the data in some fashion onto a quantum computer and stores it in a state $|\mathbf{x}\rangle$; there is a variety of techniques to do so, which e.g. stores the data in the amplitudes (coefficients) of the state, applies rotations parametrized by the data, etc. (Schuld et al., 2020; Schuld, 2021). Then, we can define a class of quantum model functions

Definition 1 (Family of Quantum Model Functions)**.** *Let $U(\theta)$ be a unitary circuit over Q qubits parametrized by $\theta \in \mathbb{R}^p$, $\mathbf{x} \in \mathbb{R}^n$ features and $S_\mathbf{x} : |0\rangle \mapsto |\mathbf{x}\rangle$ a data encoding. Then, we define a quantum model function as*

$$f_\theta^M(\mathbf{x}) = \langle \mathbf{x}| U^\dagger(\theta) M U(\theta) |\mathbf{x}\rangle\,, \tag{4}$$

for a Hermitian observable M. In order to distinguish K labels for a classification problem, we need an ensemble of size K. To that end, we define an map $R' : \mathbb{C}^{2^Q \times 2^Q} \to \mathbb{R}^K$ that describes measurement of expectation values with respect to an ensemble of K observables $\{M\}$ and storage of the respective outcomes in a K-dimensional vector. Typically, we string this together with an

https://doi.org/10.52202/079017-1004

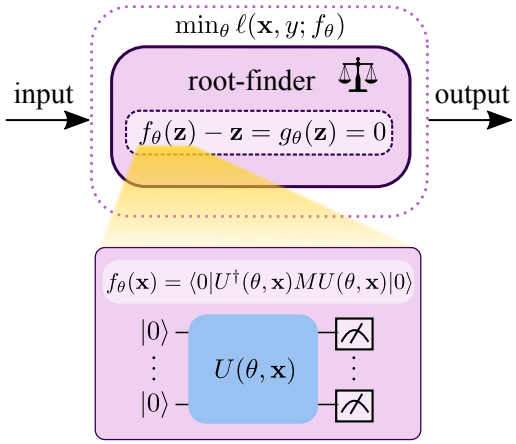


Figure 1: Instance of a Deep Equilibrium Model using a quantum model family. A black-box root-finding method is used to determine the model function's equilibrium state.

upsampling isometry $\mathcal{I}_u$ so that $R = \mathcal{I}_u \circ R'$ maps to $\mathbb{R}^n$, which allows repeated calls to reach multiple layers. This gives rise to a family of model functions $f_\theta^{\{M\}}$,

$$f_\theta^{\{M\}}(\mathbf{x}) = R\left(U(\theta)\,|\mathbf{x}\rangle\langle\mathbf{x}|\,U^\dagger(\theta)\{M\}\right). \tag{5}$$

We note that the model given in Definition 1 combined with activation functions and a bias vector was shown to be universal in Hou et al. (2023), whilst Schuld et al. (2020) provides evidence that for classification problems, as we will consider below, the architecture in Definition 1 is sufficient.

Training of such a model can proceed equivalently to the training of variational models (Bharti et al., 2022; Cerezo et al., 2021), where one defines loss functions based upon the outputs of the quantum model family $f_\theta^{\{M\}}$. While backpropagation is possible in a QML framework, it is rather uncommonly done. As was shown in Abbas et al. (2021), achieving the computational efficiency of backpropagation in classical learning models also in quantum models turns out to be very challenging and resource demanding. Most approaches to gradient evaluation thus rely on the so-called parameter-shift rule (Schuld and Killoran, 2019), where gradients can be computed exactly using an approach similar to central finite differences. The requirement for this to work is that the generator g of each unitary gate $V(\theta) = \mathrm{e}^{-\mathrm{i}\frac{\theta}{2}g}$ has eigenvalues $\pm r$. Then, $\partial_\theta f_\theta = r[f_{\theta+s} - f_{\theta-s}]$ for $s = \frac{\pi}{4r}$, where we note that for p parameters $\theta \in \mathbb{R}^p$, this gradient computation needs to be carried out for each parameter individually. Generalizations beyond two symmetric eigenvalues have been done and are in further development (Kottmann et al., 2021; Hubregtsen et al., 2022; Wierichs et al., 2022; Anselmetti et al., 2021; Kyriienko and Elfving, 2021). The gradient computation here needs to measure two expectation values per parameter up to sampling accuracy via the parameter-shift rule. For considerations on an implicit model, we expect a similar measurement overhead for the extraction of the Jacobian through a finite difference approximation in Broyden's method. When parameter shift rules are applicable, the finite differences in Broyden's method can be made exact up to sampling accuracy as well.

2.3 A Quantum Deep Equilibrium Model

We next introduce Quantum Deep Equilibrium Models, based on the notion of DEQs with the quantum model families from Definition 1. Fig. 1 depicts the overall process: In the course of minimizing a loss function, we consider the solution of a root finder of the quantum model family as the hidden layer.

While QDEQ is not restricted to classification tasks, this is what we will use as the context in our work. To perform classification, we use measurement ensembles of dimension K, as necessary for K different classes. Additionally, since the quantum basis scales exponentially in the number of qubits, we will often keep K significantly below this exponential threshold. Note that one option for

estimating these observables would be to use shadow tomography to obtain a set of classical shadows that can be used for observable estimation (Huang et al., 2020).

Given the considerations above, one choice for a measurement ensemble is a set of basis state projections $\{|k\rangle\langle k|\}_{k=1}^{K}$. This choice allows one to extract more data than the number of qubits; in fact, up to as many as the number of basis states, however this comes with a similar cost as state tomography, thus ideally we choose K well below 2^Q. Another choice of ensemble, which fits the framework of DEQs with input-injection better are Pauli measurements. This is because every Pauli measurement result falls within $[-1; 1]$ per qubit and thus directly gives a notion of perturbing the injected input towards smaller and larger. Basis state projections yield strictly positive measurement results, allowing only positive perturbations of the input injection, and tend to be smaller in magnitude as they represent probabilities. In the sense of Lloyd et al. (2020), the former corresponds to the fidelity classifier and the latter has similarities with what they call Helstrøm classifier.

We will use quantum model functions for classification into K classes and build our arguments on the following observations, which we discuss in more detail in the appendix:

(i) Such classes of models admit fixed-points, a necessary condition for DEQ frameworks to be successful. See Observation 2. The argument is based on the property how quantum models encode and extract information through measurements.

(ii) Weight-tied QDEQ with input injection is equivalent to a sequence of independently parametrized layers. See Theorem 3; extending Theorem 3 in Bai et al. (2019).

In combination, this gives evidence that the quantum model in Definition 1 with a measurement ensemble of length K can be successfully applied to classification using DEQ strategies. We are able to confirm this through our numerical experiments in the next section.

The layer depicted in Fig. 1 also contains measurement operations. As such, QDEQ does not correspond to evaluating an infinitely deep unitary quantum circuit; a layer here means quantum model including encoding and measurement. Direct application of the ideas of this paper in the context of infinitely deep circuits is hampered by both the conceptual difficulty of reasoning about fixed points for unitary isometries and the exponentially scaling effort required to extract and compare descriptions of quantum states, beyond classical shadows Huang et al. (2020).

3 Experiments

3.1 Datasets

We consider three datasets in this study. First, we consider MNIST-4, which consists of 4 classes of MNIST digits (0, 3, 6, 9) (Deng, 2012). We then extend our model to all 10 classes of MNIST (MNIST-10), as well as FashionMNIST (FashionMNIST-10) (Xiao et al., 2017). Finally, we tested our setup on natural images with CIFAR-10 Krizhevsky et al. (2009). For all datasets, we used default train/test splits[2] and randomly split the training set into 80% train, 20% validation.

3.2 Architecture

We replicate the circuit from Wang et al. (2022a) as our architecture, shown in Fig. 2. The circuit consists of four qubits. We downsample the MNIST-4 images from 28x28 to 4x4 using average pooling. For encoding, we look at the options of an angle encoding through rotation gates, as in Wang et al. (2022a), or an amplitude encoding, as implemented in `torchquantum` (Wang et al., 2022a). We pass the information through parameterized quantum gates and, finally, we measure the qubits to get a readout value from the circuit. Thus far the definition of our quantum model. Then, we transform this readout value using a small classifier head, which consists of a linear layer, then pass the result into a cross-entropy loss. As in Bai et al. (2019), we also incorporate variational dropout in the classifier head. For simulations with 10 classes, we extend this circuit by repeating the four-qubit circuit with a stride of 2 qubits in a stair-case manner, as shown in Fig. 3.

The measurement ensemble we choose for our simulations is also in alignment with Wang et al. (2022a), namely a Pauli-Z matrix per qubit, i.e. $\{M\} = \{Z \otimes I_{Q-1}, I \otimes Z \otimes I_{Q-2}, \ldots, I_{Q-1} \otimes Z\}$.

[2] `https://pytorch.org/vision/stable/datasets.html`

 https://doi.org/10.52202/079017-1004

Note that the baselines Dilip et al. (2022); Shen et al. (2024) use linear maps based on measurements of basis state projections. This introduces another layer that needs training and, most importantly, there is a risk that more general observables induce barren plateaus due to non-local support (Fontana et al., 2024; Ragone et al., 2024). Furthermore, their measurement will be increasingly costly as overlap measurement through Hadamard tests is more involved that single-qubit observables.

When it comes to data-encoding, we tested both angle and amplitude encodings for the 4 and 10 qubit experiments and generally found that the performance of the amplitude encoding is superior, both for DIRECT and IMPLICIT scenarios. Thus, we present the results for the datasets with 10 classes in the subsequent section only for amplitude encodings and show both for MNIST-4.

3.3 Models and baselines

For each dataset, we test our setup on six model variations. The first is our IMPLICIT framework. The next four are DIRECT solvers, which are weight-tied networks consisting of L repeated layers of the quantum model, where $L \in \{1, 2, 5, 10\}$. We explicitly differentiate through these layers to compare with implicit differentiation and understand whether increasing network depth results in better performance. The intuition is that if DIRECT with L repetitions shows a positive trend in performance on a task with increasing L, applying the IMPLICIT solver should be expected to be beneficial. Finally, it was found in (Bai et al., 2019, 2020) that pre-training the DEQ model using a shallow (2-layer), weight-tied DIRECT approach can considerably aid the overall convergence behaviour. This pretraining step is a cheap way to give the model reasonable weights as a starting point (called a warm-up), but the model is too shallow to achieve the best performance. We then use implicit differentiation to optimize the loss and obtain better accuracy. We call this the IMPLICIT+WARMUP solver.

In the IMPLICIT approaches, we pass the measurements from the quantum circuit into the root finder to get $\mathbf{z}^\star$, which we then pass to the classifier head and update the parameters using implicit differentiation. We also inject the original image into every iteration of the Broyden solver, in alignment with the arguments in Bai et al. (2019) and our universality theorem in Theorem 3, as input-injected weight-tied models can be phrased equivalently to models that are not weight-tied. We train the implicit models using a Broyden solver for at most 10 steps. For optimization, we use Adam (Kingma and Ba, 2014) and cross-entropy loss. We trained each model for 100 *total* epochs (i.e. if we first pre-trained using x warm-up epochs, we then trained using the implicit framework for $100 - x$ epochs) (for CIFAR-10, we only trained for 25 total epochs since we found it to converge faster). We selected hyperparameters using the validation set; see Appendix E.

Finally, for each dataset, we report the results of baselines from literature. In relation to choice of baseline, the field of quantum computational image classification is rich, with several architectures having been developed for this specific purpose, often inspired by classical convolutional neural networks (Liu et al., 2021; Henderson et al., 2020). While such models can display impressive performance on the tasks investigated here, they often employ significant classical and quantum resources in the form of classical neural networks or multiple different quantum circuit evaluations. More importantly, they represent highly specialized architectures. Since the goal of the paper is to investigate the applicability of DEQ training on general quantum models, the models chosen for this study were not highly specialized for image classification. The merit of DEQ should therefore be evaluated by comparison to results for similarly general models rather than the state-of-the-art accuracies of larger specialized models of 97% on MNIST (Henderson et al., 2020) and 83% on Fashion-MNIST (Jing et al., 2022) .

Baseline for MNIST-4 For MNIST-4, we can directly compare our results to Wang et al. (2022b) as we chose the same circuit. The difference compared to a DIRECT realization of our model with one layer is in the ultimate classification layer. The architecture can be found in Fig. 2, for more details we refer to Section 3.2.

An alternative baseline is available in West et al. (2024). For the MNIST-4 case, where they use different classes (digits 0-3) than we did, they use five qubits, compared to four in our case. Data is encoded in amplitudes. The variational circuit they use for training is built from general unitaries and they mention that they employ 20 layers.

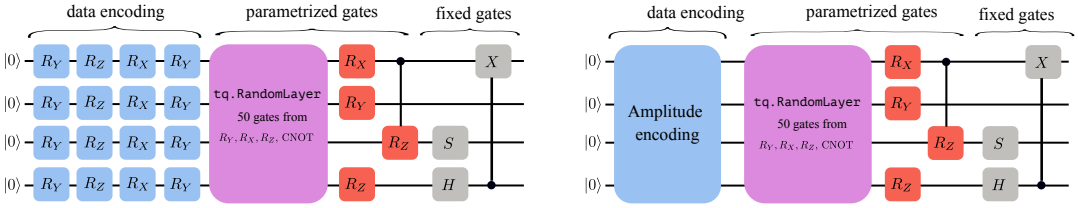
Baseline for MNIST-10 As the topic under evaluation is the concept of QDEQ training, the most applicable comparison is with the same circuit trained using DIRECT solver. However, we also include results by Alam et al. (2021) using a model of similar complexity. Their approach similarly

uses 10 qubits, as well as a similar measurement ensemble. However, beyond the training strategy, the two approaches differ in encoding strategy. While we perform a simple image scaling to a $N = 10 \times 10 = 100$ pixel image and then do $O(N)$-depth amplitude encoding in $O(\log(N))$ qubits, they do simple depth-1 angle encoding in $O(N)$ qubits, facilitated by using more complex image compression by either PCA or a trained variational autoencoder. Since they demonstrate that compression scheme has a large impact on performance (see Table 2), this makes direct comparison somewhat difficult. As for MNIST-4, West et al. (2024) is another baseline here; for MNIST with all ten classes, they used 200 layers in training, making the circuit significantly more complex than ours.

Baseline for FashionMNIST-10 For FashionMNIST, we look at the classifier circuit from Dilip et al. (2022) as a baseline because their smallest setup comes closest to our architecture, consisting of circuits of a comparable scale and number of qubits. However, the comparison is again difficult, since the results vary considerably depending on the quality of input state encoding they choose. For their encoding, quantum circuits are trained to generate approximations of amplitude-encoding-like states, called an FRQI states (Le et al., 2011), corresponding to each image. Nevertheless, high-quality FRQI states may be comparable to the amplitude encoded states used in our work. Similar work in Shen et al. (2024) additionally provide results either using amplitude encoding or using approximate FQRI and similar classifier heads to our setup. While these results use circuits with a larger number of trainable parameters (thus, likely higher expressivity), they are included for context as well. Finally, West et al. (2024) is again another baseline with equivalent setup to MNIST with all ten classes.

4 Results

In this section, we report on the result of our model on the datasets MNIST-4, MNIST-10, FashionMNIST-10, and CIFAR-10. As mentioned, all results were generated using the `torchquantum` framework (Wang et al., 2022a). Note that in the sections below, while the memory observed is in terms of classical memory that has been used for a simulation of the QML model, this is a rough measure for the number of parameters in the optimization and thus also quantifies the measurement overhead for QML models. Thus, we can expect this quantity to be a reasonable estimate, comparing different approaches relative to one another, for the necessary resources for evaluation on quantum hardware.



(a) 4x4 YZXY Angle encoding.

(b) Amplitude encoding instead of angle encoding from (a), using the implementation in `torchquantum`.

Figure 2: Circuit used for classification with up to 4 classes, following Wang et al. (2022a). Blue circuits correspond to input, purple and red shades to parametrized and trainable gates and grey to fixed gates. The RandomLayer has on average 12.5 = 50/4 two-qubit gates (CNOTs).

4.1 MNIST-4

In Table 1, we show the performance of our proposed QDEQ framework on the MNIST-4 test set. We observed that for angle encoding, deeper DIRECT models seems to perform better than shallow ones, as expected. However, the IMPLICIT models outperforms all of the DIRECT models, including the baseline. In contrast, amplitude encoding has the shallow DIRECT models perform best, and better performance in general. This may indicate both that deeper DIRECT models can suffer trainability issues compared to shallower DIRECT models, and that QDEQ does not always provide a benefit. However, as we will for MNIST-10 see below, these two effects are not linked, and need not co-occur. In addition to this, we provide a result from West et al. (2024) with a circuit of 20 layers, thus expectedly much deeper and more variational parameters than our circuit. They achieve an accuracy

https://doi.org/10.52202/079017-1004

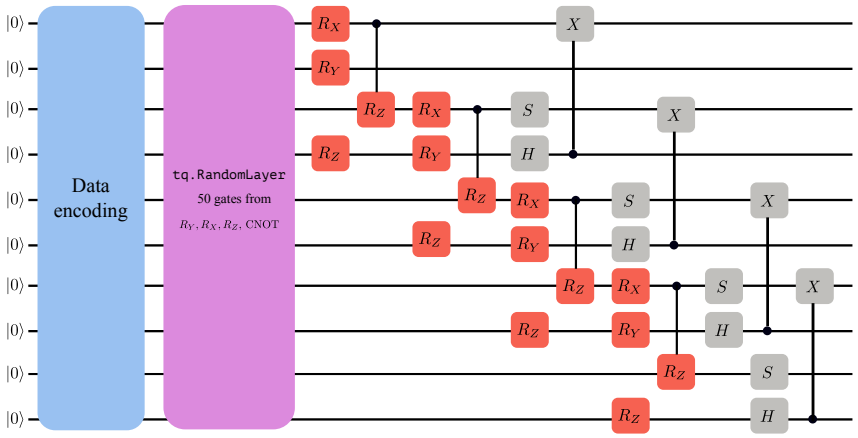


Figure 3: Extension of the circuit presented in Wang et al. (2022b) to a scenario of 10 qubits. Similarly to the tensor-network inspired circuits in Dilip et al. (2022), we repeat the four-qubit stencils in a staircase manner.

of approximately 90-92%. This supports our hypothesis that the QDEQ model is able to achieve high accuracies (up to 93.4%) at lower cost.

Table 1: Model performance on MNIST-4. Memory refers to the maximum GPU memory occupied by tensors. Runtime was calculated over 100 epochs on a NVIDIA RTX 2070 GPU. We present results both for an amplitude and angle encoding as shown in Fig. 2.

Model	Test accuracy (%)	Memory (MB)	Runtime (sec)	Residual
Amplitude Encoding				
QML Circuit [Wang et al. (2022b)] / `torchquantum` example[3]	85.3	24.49	4257.17	–
IMPLICIT solver [**ours**]	92.9	22.0	3206.43	2.395e-4
IMPLICIT+WARMUP solver [**ours**]	93.4	22.0	5276.61	6.982e-4
DIRECT solver - 10 layers [**ours**]	91.8	33.3	7820.00	9.877e-4
DIRECT solver - 5 layers [**ours**]	91.0	26.1	6013.00	0.189
DIRECT solver - 2 layers [**ours**]	92.1	21.9	3126.66	1.003
DIRECT solver - 1 layers [**ours**]	93.5	20.4	2566.00	2.867
Angle Encoding				
QML Circuit [Wang et al. (2022b)]	77.3	24.49	4762.98	–
IMPLICIT solver [**ours**]	85.8	21.23	3206.43	5.115e-4
IMPLICIT+WARMUP solver [**ours**]	86.7	22.55	2306.72	1.037e-3
DIRECT solver - 10 layers [**ours**]	84.8	39.38	3581.83	1.047
DIRECT solver - 5 layers [**ours**]	85.3	28.86	1926.27	1.818
DIRECT solver - 2 layers [**ours**]	82.7	22.55	1012.66	3.625
DIRECT solver - 1 layers [**ours**]	83.9	20.45	704.38	4.135

4.2 MNIST-10 and FashionMNIST-10

We then extend our circuit from 4 classes to 10 and show the performance on MNIST-10 (Table 2) and FashionMNIST-10 (Table 3). In both cases, the IMPLICIT framework performs better than DIRECT circuits that are 3 or more times larger in terms of memory required.

For MNIST-10, we observe that the deeper models using DIRECT training perform comparably or slightly worse than the shallowest DIRECT models. Similarly to for MNIST-4, this could indicate suboptimal training for deeper circuits. In contrast, the IMPLICIT models seem to circumvent

this problem, and slightly outperform both the DIRECT models and the baseline using PCA-based encoding. The VAE-based baseline on the other hand shows significantly higher performance, showing that such approaches to data compression (and their combination with QDEQ) may be a worthy target for further study. The results in West et al. (2024) for the all MNIST classes reach up to 79.3% (Figure 2 in their paper); while this is more accurate than ours (up to 73.68%), this can be ascribed to the vastly larger model.

For FashionMNIST-10, we observe performance differences between DIRECT and IMPLICIT models very similar to the MNIST-10 case. Furthermore, performance is competitive with the baseline results of Dilip et al. (2022), possibly reflecting a larger similarity between the encoding methods (amplitude encoding and approximate FRQI, respectively). The case for West et al. (2024) is the same as for MNIST above; with an accuracy of 74.5% (Figure 2 in their paper), they surpass our best test accuracy of 72.11%, while again, their model is far larger. Similarly, we ascribe the performance delta with Shen et al. (2024) to a similar difference in the number of trainable parameters.

Table 2: Model performance on MNIST-10. Memory refers to the maximum GPU memory occupied by tensors. Runtime was calculated over 100 epochs on a NVIDIA RTX 2070 GPU.

Model	Test accuracy (%)	Memory (MB)	Runtime (sec)	Residual
QML Circuit (VAE) [Alam et al. (2021)]	89.80	-	-	-
QML Circuit (PCA) [Alam et al. (2021)]	71.75	-	-	-
IMPLICIT solver [**ours**]	73.68	273.1	7301.74	5.598e-3
IMPLICIT+WARMUP solver [**ours**]	73.33	273.1	7017.38	1.583e-3
DIRECT solver - 10 layers [**ours**]	71.18	1233.3	2856.94	1.905e-4
DIRECT solver - 5 layers [**ours**]	72.46	628.8	1737.82	2.093e-3
DIRECT solver - 2 layers [**ours**]	72.07	266.1	4304.54	0.596
DIRECT solver - 1 layers [**ours**]	72.78	145.2	3343.40	4.026

Table 3: Model performance on FashionMNIST-10. Memory refers to the maximum GPU memory occupied by tensors. Runtime was calculated over 100 epochs on a NVIDIA RTX 2070 GPU. For Dilip et al. (2022), performance depends on encoding quality, and thus a range is given, while two comparable baselines are included for Shen et al. (2024) (see Section 3.3 for details).

Model	Test accuracy (%)	Memory (MB)	Runtime (sec)	Residual
QML Circuit [Dilip et al. (2022)]	63-75	-	-	-
QML Circuit [Shen et al. (2024)]	77-80	-	-	-
IMPLICIT solver [**ours**]	72.11	273.1	6424.67	6.892e-3
IMPLICIT+WARMUP solver [**ours**]	71.17	273.1	7240.35	1.934e-3
DIRECT solver - 10 layers [**ours**]	71.83	1233.3	11651.96	1.41e-5
DIRECT solver - 5 layers [**ours**]	70.87	628.8	6894.50	4.803e-3
DIRECT solver - 2 layers [**ours**]	70.81	266.1	4135.23	0.210
DIRECT solver - 1 layers [**ours**]	71.05	145.2	3382.09	1.431

4.3 CIFAR-10

Finally, we applied our method to CIFAR-10, a dataset of natural images; results can be found in Table 4. We find that in general, near-term quantum ML models that are amenable to numerical experiments do not perform well on CIFAR-10, as observed in recent prior work (Baek et al., 2023; Monbroussou et al., 2024). While their performance is higher than ours, we note that our models are not directly comparable. We chose not to boost model performance by adding classical NN components to focus on studying the quantum model. Still, we find that our QDEQ framework (IMPLICIT+WARMUP) has higher accuracy on the test set than the direct solver baseline. This motivates the utility of our method on more realistic datasets.

https://doi.org/10.52202/079017-1004

Table 4: Model performance on CIFAR-10.

Model	Test accuracy (%)
IMPLICIT solver [**ours**]	24.38
IMPLICIT+WARMUP solver [**ours**]	**25.45**
DIRECT solver - 10 layers [**ours**]	23.71
DIRECT solver - 5 layers [**ours**]	24.19
DIRECT solver - 2 layers [**ours**]	24.90
DIRECT solver - 1 layers [**ours**]	24.70

5 Conclusion and open problems

In this work, we introduce a novel approach to training QML models under the framework of deep equilibrium models. We theoretically demonstrate that under certain conditions, QML models for classification tasks are amenable to DEQ-based optimization. This finding aligns with our numerical observations, suggesting that DEQs can significantly reduce circuit complexity while maintaining or improving performance compared to explicit training methods; this benefit is expected to hold in particular compared to using deeper parametrized circuits in quantum neural networks to achieve similar expressibility to the QDEQ model. This is particularly advantageous for near-term quantum algorithms, where minimizing circuit depth is crucial.

Our numerical experiments indicate that, for the case of quantum model families with theoretical intuition about their fixed-point properties, the application of an implicit QDEQ model training is superior compared to a set of explicitly repeated layers in almost all cases, with the exception of amplitude-encoded data on our experiment with only four classes. Generally, our experiments indicate that the accuracy of our model is competitive with baseline models that typically require more classical and quantum resources.

One of the key benefits of the DEQ approach is that it avoids the computationally expensive differentiation through the root finder. However, measurements are still required during the forward pass to guide the root finder's Broyden-based updates. The cost of these measurements is likely comparable to the parameter-shift rule commonly used in gradient-based optimization. Further investigation is needed to quantify the measurement overhead and impact on the optimization process. Additionally, the influence of shot noise and other quantum noise sources on DEQ-based training remains to be explored, the expectation being that the impact of noise on DEQ training will behave similarly to general QNN training. Understanding the training of quantum models that are often amenable to vanishing gradients, called barren plateaus, has been greatly advanced by some recent theory works (Fontana et al., 2024; Ragone et al., 2024). We anticipate QDEQ to behave similarly with respect to vanishing gradients as general VQAs. More in-depth analysis is an open problem. The potential advantage of DEQ here is the option of using shallower circuits for a given performance, which are known to be less prone to barren plateaus.

Another avenue of further research could be considering the outcomes of a preceding quantum experiment as quantum data. This could be a physical experiment or a preceding quantum algorithm. Either can be seen as an advanced encoding map. Further investigations towards this approach, and into whether any advantages transfer, is left for further research. Finally, it would be interesting to extend the applicability of DEQs in combination with our deliberations in Observation 2 to a broader range of QML models.

Acknowledgements

We thank Zachary Cetinic for his insights and rooting for the ultimate side-quest. AAG thanks Anders G. Frøseth for his generous support and acknowledges the generous support of Natural Resources Canada and the Canada 150 Research Chairs program. Parts of the resources used in preparing this research were provided by the Province of Ontario, the Government of Canada through CIFAR, and companies sponsoring the Vector Institute. LBK acknowledges support from the Carlsberg Foundation, grant CF-21-0669. RAVH acknowledges NSERC Discovery Grant No. RGPIN-2024-06594.

References

Scott Aaronson. 2015. Read the fine print. Nature Physics 11, 4 (2015), 291–293. https://doi.org/10.1038/nphys3272

Amira Abbas, David Sutter, Christa Zoufal, Aurelien Lucchi, Alessio Figalli, and Stefan Woerner. 2021. The power of quantum neural networks. Nature Computational Science 1, 6 (2021), 403–409. https://doi.org/10.1038/s43588-021-00084-1

Shahnawaz Ahmed, Nathan Killoran, and Juan Felipe Carrasquilla Álvarez. 2022. Implicit differentiation of variational quantum algorithms. arXiv:2211.13765 [quant-ph] (2022). https://arxiv.org/abs/2211.13765

Mahabubul Alam, Satwik Kundu, Rasit Onur Topaloglu, and Swaroop Ghosh. 2021. Quantum-classical hybrid machine learning for image classification (iccad special session paper). In 2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD). IEEE, 1–7. https://ieeexplore.ieee.org/abstract/document/9643516

Gian-Luca R. Anselmetti, David Wierichs, Christian Gogolin, and Robert M. Parrish. 2021. Local, expressive, quantum-number-preserving VQE ansätze for fermionic systems. New Journal of Physics 23, 11 (2021), 113010. https://doi.org/10.1088/1367-2630/ac2cb3

Hankyul Baek, Soohyun Park, and Joongheon Kim. 2023. Logarithmic dimension reduction for quantum neural networks. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. 3738–3742. https://dl.acm.org/doi/10.1145/3583780.3615240

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. 2019. Deep Equilibrium Models. Advances in Neural Information Processing Systems 32 (2019). https://proceedings.neurips.cc/paper/2019/hash/01386bd6d8e091c2ab4c7c7de644d37b-Abstract.html

Shaojie Bai, Vladlen Koltun, and J Zico Kolter. 2020. Multiscale Deep Equilibrium Models. Advances in Neural Information Processing Systems 33 (2020), 5238–5250. https://proceedings.neurips.cc/paper_files/paper/2020/hash/3812f9a59b634c2a9c574610eaba5bed-Abstract.html

Shaojie Bai, Vladlen Koltun, and J. Zico Kolter. 2021. Stabilizing Equilibrium Models by Jacobian Regularization. arXiv:2106.14342 [cs, stat] (2021). http://arxiv.org/abs/2106.14342 arXiv: 2106.14342.

Kerstin Beer, Dmytro Bondarenko, Terry Farrelly, Tobias J Osborne, Robert Salzmann, Daniel Scheiermann, and Ramona Wolf. 2020. Training deep quantum neural networks. Nature Communications 11, 1 (2020), 808. https://www.nature.com/articles/s41467-020-14454-2

Kishor Bharti, Alba Cervera-Lierta, Thi Ha Kyaw, Tobias Haug, Sumner Alperin-Lea, Abhinav Anand, Matthias Degroote, Hermanni Heimonen, Jakob S Kottmann, Tim Menke, , Wai-Keong Mok, Sukin Sim, Leong-Chuan Kwek, and Alán Aspuru-Guzik. 2022. Noisy intermediate-scale quantum algorithms. Reviews of Modern Physics 94, 1 (2022), 015004. https://doi.org/10.1103/RevModPhys.94.015004

Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. 2017. Quantum machine learning. Nature 549, 7671 (2017), 195–202. https://doi.org/10.1038/nature23474

Charles G. Broyden. 1965. A class of methods for solving nonlinear simultaneous equations. Math. Comp. 19, 92 (1965), 577–593. https://doi.org/10.1090/s0025-5718-1965-0198670-6

Matthias C Caro, Hsin-Yuan Huang, Marco Cerezo, Kunal Sharma, Andrew Sornborger, Lukasz Cincio, and Patrick J Coles. 2022. Generalization in quantum machine learning from few training data. Nature communications 13, 1 (2022), 4919. https://www.nature.com/articles/s41467-022-32550-3

M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. 2021. Variational quantum algorithms. Nature Reviews Physics 3, 9 (2021), 625–644. https://doi.org/10.1038/s42254-021-00348-9

Haoyu Chu, Shikui Wei, and Ting Liu. 2023. Lyapunov-Stable Deep Equilibrium Models. arXiv:2304.12707 [cs] (2023). https://arxiv.org/abs/2304.12707

Li Deng. 2012. The mnist database of handwritten digit images for machine learning research. IEEE Signal Processing Magazine 29, 6 (2012), 141–142. http://yann.lecun.com/exdb/mnist/

Rohit Dilip, Yu-Jie Liu, Adam Smith, and Frank Pollmann. 2022. Data compression for quantum machine learning. Physical Review Research 4, 4 (2022), 043007. https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.4.043007

Enrico Fontana, Dylan Herman, Shouvanik Chakrabarti, Niraj Kumar, Romina Yalovetzky, Jamie Heredge, Shree Hari Sureshbabu, and Marco Pistoia. 2024. Characterizing barren plateaus in quantum ansätze with the adjoint representation. Nature Communications 15, 1 (2024), 7171. https://doi.org/10.1038/s41467-024-49910-w

Zhengyang Geng and J Zico Kolter. 2023. Torchdeq: A library for deep equilibrium models. arXiv:2310.18605 [cs] (2023). https://arxiv.org/abs/2310.18605

Maxwell Henderson, Samriddhi Shakya, Shashindra Pradhan, and Tristan Cook. 2020. Quanvolutional neural networks: powering image recognition with quantum circuits. Quantum Machine Intelligence 2, 1 (2020), 2. https://link.springer.com/article/10.1007/s42484-020-00012-y

Xiaokai Hou, Guanyu Zhou, Qingyu Li, Shan Jin, and Xiaoting Wang. 2023. A duplication-free quantum neural network for universal approximation. Science China Physics, Mechanics & Astronomy 66, 7 (2023), 270362. https://link.springer.com/article/10.1007/s11433-023-2098-8

Hsin-Yuan Huang, Richard Kueng, and John Preskill. 2020. Predicting many properties of a quantum system from very few measurements. Nature Physics 16, 10 (2020), 1050–1057. https://www.nature.com/articles/s41567-020-0932-7

Thomas Hubregtsen, Frederik Wilde, Shozab Qasim, and Jens Eisert. 2022. Single-component gradient rules for variational quantum algorithms. Quantum Science and Technology 7, 3 (2022), 035008. https://doi.org/10.1088/2058-9565/ac6824

Yu Jing, Xiaogang Li, Yang Yang, Chonghang Wu, Wenbing Fu, Wei Hu, Yuanyuan Li, and Hua Xu. 2022. RGB image classification with quantum convolutional ansatz. Quantum Information Processing 21, 3 (2022), 101. https://link.springer.com/article/10.1007/s11128-022-03442-8

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. arXiv:1412.6980 [cs] (2014). https://arxiv.org/abs/1412.6980

Jakob S. Kottmann, Abhinav Anand, and Alán Aspuru-Guzik. 2021. A feasible approach for automatically differentiable unitary coupled-cluster on quantum computers. Chemical Science 12, 10 (2021), 3497–3508. https://doi.org/10.1039/D0SC06627C

Steven G. Krantz and Harold R. Parks. 2013. Basic Ideas. In The Implicit Function Theorem: History, Theory, and Applications, Steven G. Krantz and Harold R. Parks (Eds.). New York, NY, 35–59. https://doi.org/10.1007/978-1-4614-5981-1_3

Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009). https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

Oleksandr Kyriienko and Vincent E. Elfving. 2021. Generalized quantum circuit differentiation rules. Physical Review A 104, 5 (2021), 052417. https://doi.org/10.1103/PhysRevA.104.052417

Phuc Q. Le, Fangyan Dong, and Kaoru Hirota. 2011. A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. Quantum Information Processing 10 (2011), 63–84. https://doi.org/10.1007/s11128-010-0177-y

Mingjie Li, Yisen Wang, and Zhouchen Lin. 2022. Cerdeq: Certifiable deep equilibrium model. In International Conference on Machine Learning. PMLR, 12998–13013. https://proceedings.mlr.press/v162/li22t.html

Junhua Liu, Kwan Hui Lim, Kristin L Wood, Wei Huang, Chu Guo, and He-Liang Huang. 2021. Hybrid quantum-classical convolutional neural networks. Science China Physics, Mechanics & Astronomy 64, 9 (2021), 290311. https://link.springer.com/article/10.1007/s11433-021-1734-3

Seth Lloyd, Maria Schuld, Aroosa Ijaz, Josh Izaac, and Nathan Killoran. 2020. Quantum embeddings for machine learning. arXiv:2001.03622 [quant-ph] (2020). http://arxiv.org/abs/2001.03622 arXiv: 2001.03622.

Jarrod R. McClean, Sergio Boixo, Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven. 2018. Barren plateaus in quantum neural network training landscapes. Nature Communications 9, 1 (2018), 4812. https://doi.org/10.1038/s41467-018-07090-4

Léo Monbroussou, Jonas Landman, Letao Wang, Alex B. Grilo, and Elham Kashefi. 2024. Subspace Preserving Quantum Convolutional Neural Network Architectures. https://arxiv.org/abs/2409.18918

Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'brien. 2014. A variational eigenvalue solver on a photonic quantum processor. Nature Communications 5, 1 (2014), 4213. https://www.nature.com/articles/ncomms5213

Lirandë Pira and Chris Ferrie. 2024. On the interpretability of quantum neural networks. Quantum Machine Intelligence 6, 2 (2024), 52. https://link.springer.com/article/10.1007/s42484-024-00191-y

Ashwini Pokle, Zhengyang Geng, and J Zico Kolter. 2022. Deep equilibrium approaches to diffusion models. Advances in Neural Information Processing Systems 35 (2022), 37975–37990. https://proceedings.neurips.cc/paper_files/paper/2022/hash/f7f47a73d631c0410cbc2748a8015241-Abstract-Conference.html

Michael Ragone, Bojko N Bakalov, Frédéric Sauvage, Alexander F Kemper, Carlos Ortiz Marrero, Martín Larocca, and M Cerezo. 2024. A Lie algebraic theory of barren plateaus for deep parameterized quantum circuits. Nature Communications 15, 1 (2024), 7172. https://doi.org/10.1038/s41467-024-49909-3

Maria Schuld. 2021. Supervised quantum machine learning models are kernel methods. arXiv:2101.11020 [quant-ph, stat] (2021). https://arxiv.org/abs/2101.11020v2

Maria Schuld, Alex Bocharov, Krysta M Svore, and Nathan Wiebe. 2020. Circuit-centric quantum classifiers. Physical Review A 101, 3 (2020), 032308. https://journals.aps.org/pra/abstract/10.1103/PhysRevA.101.032308

Maria Schuld and Nathan Killoran. 2019. Quantum machine learning in feature Hilbert spaces. Physical review letters 122, 4 (2019), 040504. https://doi.org/10.1103/PhysRevLett.122.040504

Kevin Shen, Bernhard Jobst, Elvira Shishenina, and Frank Pollmann. 2024. Classification of the Fashion-MNIST Dataset on a Quantum Computer. arXiv:2403.02405 [quant-ph, cs] (2024). https://arxiv.org/abs/2403.02405

Hanrui Wang, Yongshan Ding, Jiaqi Gu, Yujun Lin, David Z Pan, Frederic T Chong, and Song Han. 2022a. QuantumNAS: Noise-adaptive search for robust quantum circuits. In 2022 IEEE International Symposium on High-Performance Computer Architecture (HPCA). IEEE, 692–708. https://doi.org/10.1109/HPCA53966.2022.00057

Hanrui Wang, Zirui Li, Jiaqi Gu, Yongshan Ding, David Z Pan, and Song Han. 2022b. QOC: quantum on-chip training with parameter shift and gradient pruning. In Proceedings of the 59th ACM/IEEE Design Automation Conference. 655–660. https://dl.acm.org/doi/abs/10.1145/3489517.3530495

Maxwell T. West, Azar C. Nakhl, Jamie Heredge, Floyd M. Creevey, Lloyd C. L. Hollenberg, Martin Sevior, and Muhammad Usman. 2024. Drastic Circuit Depth Reductions with Preserved Adversarial Robustness by Approximate Encoding for Quantum Machine Learning. Intelligent Computing 3 (2024). https://doi.org/10.34133/icomputing.0100

David Wierichs, Josh Izaac, Cody Wang, and Cedric Yen-Yu Lin. 2022. General parameter-shift rules for quantum gradients. Quantum 6 (2022), 677. https://doi.org/10.22331/q-2022-03-30-677

Ezra Winston and J Zico Kolter. 2020. Monotone operator equilibrium networks. Advances in neural information processing systems 33 (2020), 10718–10728. https://proceedings.neurips.cc/paper/2020/hash/798d1c2813cbdf8bcdb388db0e32d496-Abstract.html

Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. arXiv:1708.07747 [cs] (2017). https://arxiv.org/abs/1708.07747

Appendix

A Existence of fixed points for quantum model families

Observation 2 (Contractiveness of Quantum Model Families)**.**

Below, we provide a discussion regarding the existence of fixed points for quantum model functions according to Definition 1, with the additional assumption that measurement ensembles only consist of Pauli operators and projectors on computational basis states.

Consider two vectors $\mathbf{z}, \mathbf{z}'$, which, without loss of generality, we may assume to be normalized over $[0; 2\pi]^n$. Further, consider a Hermitian observable M used for readout. The value of this readout for quantum models as in Definition 1 then takes the form

$$\mathrm{Tr}\left(MUS_{\mathbf{z}}\left|0\right\rangle\!\left\langle 0\right|S_{\mathbf{z}}^{\dagger}U^{\dagger}\right). \tag{6}$$

Now, if we want to evaluate contractiveness of the quantum model and consider

$$\left\|f^{\{M\}}(\mathbf{z}) - f^{\{M\}}(\mathbf{z}')\right\| \tag{7}$$

for a family of quantum models, we can apply the triangle inequality to arrive at conditions for each entry corresponding to a single observable of the ensemble. Thus, to argue whether the model is contractive, we need to bound quantities of the form

$$\Delta = \left|\mathrm{Tr}\left(MUS_{\mathbf{z}}\left|0\right\rangle\!\left\langle 0\right|S_{\mathbf{z}}^{\dagger}U^{\dagger}\right) - \mathrm{Tr}\left(MUS_{\mathbf{z}'}\left|0\right\rangle\!\left\langle 0\right|S_{\mathbf{z}'}^{\dagger}U^{\dagger}\right)\right|. \tag{8}$$

We use notation as before and say $S_{\mathbf{z}}\left|0\right\rangle = \left|z\right\rangle$. Note that due to the cyclic properties of the trace, we can construct an equivalent observable $\tilde{M} = U^{\dagger}MU$ with the same spectrum as M, and write this as

$$\Delta = \left|\mathrm{Tr}\left(\tilde{M}\left(\left|\mathbf{z}\right\rangle\!\left\langle\mathbf{z}\right| - \left|\mathbf{z}'\right\rangle\!\left\langle\mathbf{z}'\right|\right)\right)\right|. \tag{9}$$

We furthermore can express $\tilde{M}$ in its eigenbasis, $\tilde{M} = \sum_{\lambda} M_{\lambda}\left|\lambda\right\rangle\!\left\langle\lambda\right|$. Then evaluating the trace in this basis allows us to simplify the expression as

$$\Delta = \Big|\sum_{\lambda} M_{\lambda}\big(\left|\langle\lambda|\mathbf{z}\rangle\right|^2 - \left|\langle\lambda|\mathbf{z}'\rangle\right|^2\big)\Big|. \tag{10}$$

Now we assume that all eigenvalues of M are bounded, i.e., that it has a finite spectral norm $\|M\| < \infty$. This spectral bound and another application of the triangle inequality yields

$$\Delta \leq \|M\| \sum_{\lambda} \Big|\left|\langle\lambda|\mathbf{z}\rangle\right|^2 - \left|\langle\lambda|\mathbf{z}'\rangle\right|^2\Big|, \tag{11}$$

where the right hand side is the total variation distance with respect to some POVM. As such, it is bounded by the trace norm. Then, we can conclude that

$$\Delta \leq 2\|M\| \; \left\|\left|\mathbf{z}\right\rangle\!\left\langle\mathbf{z}\right| - \left|\mathbf{z}'\right\rangle\!\left\langle\mathbf{z}'\right|\right\|_{\mathrm{Tr}} = 2\|M\|\sqrt{1 - \left|\langle\mathbf{z}|\mathbf{z}'\rangle\right|^2}. \tag{12}$$

As detailed in Appendix C, this bound can be further sharpened to not require the factor of two in the case of basis-state measurements, i.e.,

$$\Delta \leq \|M\|\sqrt{1 - \left|\langle\mathbf{z}|\mathbf{z}'\rangle\right|^2}. \tag{13}$$

For further discussion of the general case, please see the end of this appendix. Note that in the case of a single observable, this expressions is clearly bounded by $\|M\|$. Assuming that this is smaller than or equal to one, which is clearly a necessary assumption for a contraction map to be possible, we therefore have that the distances after application of the model are always less than one, meaning for any pair of vectors such that $\|\mathbf{z} - \mathbf{z}'\| > 1$, a map of this is always subcontractive. Thus, for the case of a single observable M, we only need to consider the case where $\|\mathbf{z} - \mathbf{z}'\| \leq 1$.

Assume that the following holds for some $c > 0$:

$$\left|\langle\mathbf{z}|\mathbf{z}'\rangle\right| \geq 1 - \frac{c}{2}\sin\left(\left\|\mathbf{z} - \mathbf{z}'\right\|^2\right). \tag{14}$$

https://doi.org/10.52202/079017-1004

Evidence for this bound with $c = 1$ in the case of amplitude encoding can be found in Appendix B, along with evidence for a slightly weaker $c = 2$ bound for the specific angle encoding strategy used in this work. From the inequality, we have the following:

$$1 - |\langle \mathbf{z}|\mathbf{z}'\rangle|^2 \leq 1 - \left(1 - \frac{c}{2}\sin\left(\|\mathbf{z} - \mathbf{z}'\|^2\right)\right)^2 \tag{15}$$

$$= c\sin\left(\|\mathbf{z} - \mathbf{z}'\|^2\right) - \frac{c^4}{4}\sin\left(\|\mathbf{z} - \mathbf{z}'\|^2\right)^2 \tag{16}$$

$$\leq c\sin\left(\|\mathbf{z} - \mathbf{z}'\|^2\right) \leq c\,\|\mathbf{z} - \mathbf{z}'\|^2 \ , \tag{17}$$

with the final result that

$$|f(\mathbf{z}) - f(\mathbf{z}')| \leq c\,\|M\|\|\mathbf{z} - \mathbf{z}'\| \tag{18}$$

for a single output variable so that $\|M\| \leq 1$. Note that the constant c appears as an overall scale in the expression. For simplicity, we will focus on the amplitude encoding case of $c = 1$ below—See Appendix B.2 for a discussion of how the angle-encoding case differs.

Having bounded a single observable, we next discuss how to deal with multiple observables in a family of model functions. If we have an ensemble for K observables, $\{M_k\}_{k=1}^K$, then by the properties of the ℓ_2 norm,

$$\left\|f^{\{M\}}(\mathbf{z}) - f^{\{M\}}(\mathbf{z}')\right\|_{\ell_2} \leq \sqrt{\textstyle\sum_k \|M_k\|^2}\|\mathbf{z} - \mathbf{z}'\|_{\ell_2}. \tag{19}$$

So if all $\|M_k\|$ are bounded by 1, we will have a prefactor of $\sqrt{K}$ for K observables, which suggests that for more than one observable, the model family is not (sub)contractive in the ℓ_2 norm. However, the case would be different if we choose the maximum-norm as a metric. Then, we do not encounter the usual property of the ℓ_2 norm that it grows with the square-root of the dimension, and we instead have

$$\left\|f^{\{M\}}(\mathbf{z}) - f^{\{M\}}(\mathbf{z}')\right\|_{\max} \leq \max_k \|M_k\|\|\mathbf{z} - \mathbf{z}'\|_{\max}. \tag{20}$$

This means, that our model (family) f from Definition 1 is subcontractive, which is not sufficient yet for a fixed point to exist. One way to proceed would be to show that

$$\|f(f(\mathbf{z})) - f(\mathbf{z})\| < \|f(\mathbf{z}) - \mathbf{z}\| \tag{21}$$

except for $f(\mathbf{z}) = \mathbf{z}$, i.e. unless $\mathbf{z}$ is a fixed point.

Instead, we make make a different deliberation. We restrict ourselves to observables that are either Pauli-operations, which are unitary and have eigenvalues ± 1, or projectors on basis states, whose $+1$ eigenspace is spanned by a single basis state only, as they are rank-one. Note than the bound we used above in Eq. (11) can be loose, as it assumes a worst-case where the difference is aligned in eigenspaces so that $|\mathrm{Tr}(M(|\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'|))| = 2\|M\|\||\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'|\|_{\mathrm{Tr}}$ is attained. Such a scenario is highly unlikely. In fact, consider the relationship in Eq. (21), and denote $|f(\mathbf{z})\rangle = S_{f(\mathbf{z})}\,|0\rangle$. Then, we look at

$$\|f(f(\mathbf{z})) - f(\mathbf{z})\| = \left|\ \mathrm{Tr}\left(M(|f(\mathbf{z})\rangle\langle f(\mathbf{z})| - |\mathbf{z}\rangle\langle\mathbf{z}|)\right)\right|. \tag{22}$$

By the arguments of Appendix C, a quantum model is then not contractive if the difference $|f(\mathbf{z})\rangle\langle f(\mathbf{z})| - |\mathbf{z}\rangle\langle\mathbf{z}|$ is fully aligned with either the $+1$ and -1 eigenspaces for Pauli operators or the $+1$ eigenspace of a basis state projection, in the sense outlined in that appendix. Surely, as $\mathbf{z}$ comes from a batch of data with different values for each element, such a construction would either not correspond to a sensible set of observables or the encoding would not capture any diversity in the data. Thus, we can expect that the quantum models on data will act as contractions in practice with high probability. This also likely explains why the architectures not fully covered by the analysis above (i.e., angle encoding and Pauli measurements) work in practice.

For the case of amplitude encoding, a re-normalization of the outputs would be required before re-encoding them. Viewing this necessary rescaling of $f^{\{M\}}(\mathbf{z})$ as part of the map may result in a change in norms, since in general $\|\mathbf{x} - \mathbf{y}\| \neq \left\|\frac{\mathbf{x}}{\|\mathbf{x}\|} - \frac{\mathbf{y}}{\|\mathbf{y}\|}\right\|$. Given the positive empirical results of the main text, we conjecture that the arguments for strict contractivity put forth above in most cases more than compensates for any such shifts; we leave further investigation as a topic for future work.

These discussions identify two key aspects to achieve contractive quantum architectures of the form considered in this work: The encoding should encode similar vectors into similar quantum states; and, the measurement operators should not amplify these differences when converting back into classical data. The interplay of these two aspects is captured well by Eq. (9), which highlights the simple fact that the difference in output depends on the difference in the encoded states and the ability of the measurement to resolve this difference. Our approach here used the bound in Eq. (14) as a way to quantify that the encoding preserves closeness, and bounded the resolving/amplification power of the readout using simple operator norms, leaving more detailed analysis of the interplay between encoding and readout as future work.

B Evidence for overlap bounds

In Appendix A, the following property for the encoding of vectors fulfilling $\|\mathbf{z} - \mathbf{z}'\| \leq 1$ was used:

$$|\langle \mathbf{z}|\mathbf{z}'\rangle| \geq 1 - \frac{c}{2}\sin\left(\|\mathbf{z} - \mathbf{z}'\|^2\right). \tag{23}$$

In this section, we provide evidence for this bound for the two encoding maps used in the main text, i.e., amplitude encoding and angle encoding.

B.1 Amplitude encoding

This is the simplest case. Assuming that the inputs are normalized to length one, and using that we only work with real vectors, the following holds by definition of amplitude encoding:

$$|\langle \mathbf{z}|\mathbf{z}'\rangle| = |\mathbf{z}\cdot\mathbf{z}'| \ . \tag{24}$$

Furthermore,

$$\|\mathbf{z} - \mathbf{z}'\|^2 = \|\mathbf{z}\|^2 + \|\mathbf{z}'\|^2 - 2\mathbf{z}\cdot\mathbf{z}' \tag{25}$$

$$= 2 - 2\mathbf{z}\cdot\mathbf{z}' \ , \tag{26}$$

meaning

$$|\langle \mathbf{z}|\mathbf{z}'\rangle| = \left|1 - \frac{1}{2}\|\mathbf{z} - \mathbf{z}'\|^2\right|. \tag{27}$$

Using the fact that the norm in this expression is bounded by one, we therefore get

$$|\langle \mathbf{z}|\mathbf{z}'\rangle| = 1 - \frac{1}{2}\|\mathbf{z} - \mathbf{z}'\|^2 \tag{28}$$

$$\geq 1 - \frac{1}{2}\sin\left(\|\mathbf{z} - \mathbf{z}'\|^2\right), \tag{29}$$

as desired. Note that for this particular encoding, the equality in Eq. (28) allows for the introduction of a factor $\frac{1}{2}$ in Eq. (17), corresponding to the the constant $c = 1$ in the bound in Eq. (23).

B.2 Angle encoding

In the case of the angle encoding, consider the unitaries used in the encoding:

$$|\langle \mathbf{z}|\mathbf{z}'\rangle| = \left|\langle 0| S_{\mathbf{z}}^{\dagger} S_{\mathbf{z}'} |0\rangle\right|. \tag{30}$$

For the specific encoding used, these unitaries have the property that they consist of single-qubit encodings operating on separate qubits. In other words, denoting these single-qubit maps by $S_{\mathbf{z}}^{(1)}$ and splitting the vectors into their constituent 4-tuples of entries,

$$\mathbf{z}_k = \left[(\mathbf{z})_{4k+1}, (\mathbf{z})_{4k+2}, (\mathbf{z})_{4k+3}, (\mathbf{z})_{4k+4}\right]^T, \tag{31}$$

we can write the encoding map as the following tensor product

$$S_{\mathbf{z}} = \bigotimes_{k=1}^{Q} S_{\mathbf{z}_k}^{(1)}. \tag{32}$$

https://doi.org/10.52202/079017-1004

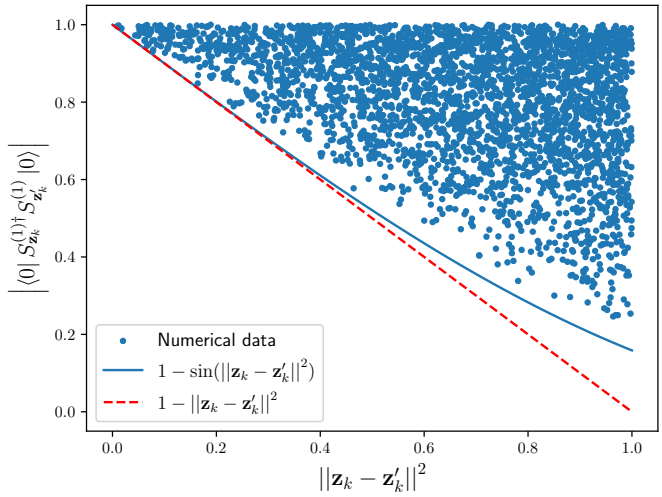


Figure 4: Plots of the relation between the single-qubit overlaps $\left|\langle 0| S^{(1)\dagger}_{\mathbf{z}_k} S^{(1)}_{\mathbf{z}'_k} |0\rangle\right|$ and the norm $\|\mathbf{z}_k - \mathbf{z}'_k\|^2$ for 3000 pairs of random vectors in $\mathbb{R}^4$. Note that all of the points lie above the line corresponding to $1 - \sin\left(\|\mathbf{z}_k - \mathbf{z}'_k\|^2\right)$.

This implies that the overlap takes the form:

$$|\langle \mathbf{z}|\mathbf{z}'\rangle| = \prod_{k=1}^{Q} \left|\langle 0| S^{(1)\dagger}_{\mathbf{z}_k} S^{(1)}_{\mathbf{z}'_k} |0\rangle\right|. \tag{33}$$

Each factor in this expression is in principle simply a complicated trigonometric expression in eight variables. While it is likely possible to bound this expression analytically, a simple numerical investigation was assumed sufficient evidence for our investigation here; see Fig. 4. Drawing 3000 pairs of vectors $\mathbf{z}_k, \mathbf{z}'_k \in \mathbb{R}^4$ at random so that $\|\mathbf{z}_k - \mathbf{z}'_k\| \leq 1$, we see that in all cases the following inequality holds,

$$\left|\langle 0| S^{(1)\dagger}_{\mathbf{z}_k} S^{(1)}_{\mathbf{z}'_k} |0\rangle\right| \geq 1 - \sin\left(\|\mathbf{z}_k - \mathbf{z}'_k\|^2\right). \tag{34}$$

To finish the bound, we apply the following trigonometric identity,

$$\sin(a+b) = \sin(a)\cos(b) + \sin(b)\cos(a) \tag{35}$$

and use this to derive:

$$\begin{aligned}
(1-\sin(a))(1-\sin(b)) &= 1 - \sin(a) - \sin(b) - 2\sin(a)\sin(b) && (36)\\
&= 1 - \sin(a) - \sin(b) - 2\sin(a)\sin(b) && (37)\\
&\quad + \sin(a)\cos(b) + \sin(b)\cos(a) - \sin(a+b) && (38)\\
&= 1 - \sin(a+b) && (39)\\
&\quad + \sin(a)(\cos(b) + \sin(b) - 1) && (40)\\
&\quad + \sin(b)(\cos(a) + \sin(a) - 1). && (41)
\end{aligned}$$

For $0 < a, b < 1$, the two final terms are either positive or zero. Thus, under this assumption on the arguments, we obtain

$$(1-\sin(a))(1-\sin(b)) \geq 1 - \sin(a+b). \tag{42}$$

Combining Eq. (33) with the bounds of Eq. (34) and Eq. (42) now yields

$$|\langle \mathbf{z}|\mathbf{z}'\rangle| = \prod_{k=1}^{Q} \left| \langle 0| S_{\mathbf{z}_k}^{(1)\dagger} S_{\mathbf{z}'_k}^{(1)} |0\rangle \right| \tag{43}$$

$$\geq \prod_{k=1}^{Q} \left(1 - \sin\left(\|\mathbf{z}_k - \mathbf{z}'_k\|^2\right)\right) \tag{44}$$

$$\geq 1 - \sin\left(\sum_k \|\mathbf{z}_k - \mathbf{z}'_k\|^2\right) \tag{45}$$

$$= 1 - \sin\left(\|\mathbf{z} - \mathbf{z}'\|^2\right), \tag{46}$$

as desired.

Note that, in contrast to the amplitude-encoding case, we do not have a pre-factor of $1/2$ on the second term. In other words, our bound corresponds to the one in Eq. (23) with $c = 2$. As can be seen by the derivations in Appendix A, this introduces an overall factor of 2 that would in principle need to be compensated for, e.g., in the magnitude of the readout operators. However, our experiments indicate no problems finding fixed points also without such adaptations. We ascribe this success to the likely looseness of the bounds derived in Appendix A, since this means contractiveness can be present even when the derived bounds are not sufficient to guarantee it. For a further discussion of some of the sources of this looseness, see Appendix A.

C Further analysis of bound tightness

In this section, we provide a more detailed analysis of the steps leading from Eq. (9) to Eq. (12) in Appendix A, including the derivation of a tighter bound for the case of observables that are projectors.

Consider the object to be bounded,

$$\Delta = \left| \mathrm{Tr}\left(\tilde{M}\left(|\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'|\right)\right)\right|. \tag{47}$$

Looking more closely at the object $|\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'|$, we can note that this is a Hermitian, traceless rank-2 operator. This implies that it can be diagonalized unitarily, with at most two nonzero eigenvalues that necessarily sum to zero due to the tracelessness. In other words, it can be written on the form

$$|\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'| = \lambda\left(|\xi_0\rangle\langle\xi_0| - |\xi_1\rangle\langle\xi_1|\right) \tag{48}$$

for some $\lambda \geq 0$. In fact, an explicit calculation shows that this constant can be characterized as $\lambda = \||\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'|\|_{\mathrm{Tr}}$. Thus, Eq. (47) can be rewritten as

$$\Delta = \left|\mathrm{Tr}\Big(\tilde{M}\left(|\xi_0\rangle\langle\xi_0| - |\xi_1\rangle\langle\xi_1|\right)\Big)\right| \ \||\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'|\|_{\mathrm{Tr}} \tag{49}$$

$$= \left|\mathrm{Tr}\left(\tilde{M}\,|\xi_0\rangle\langle\xi_0|\right) - \mathrm{Tr}\left(\tilde{M}\,|\xi_1\rangle\langle\xi_1|\right)\right| \ \||\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'|\|_{\mathrm{Tr}}. \tag{50}$$

Comparing this to the bound in Eq. (12),

$$\Delta \leq 2\|M\| \ \||\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'|\|_{\mathrm{Tr}}, \tag{51}$$

it becomes clear that the tightness of the bound depends on how effectively the observable $\tilde{M}$ distinguishes between the two eigenstates of $|\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'|$. This, in turn, depends on the alignment of these eigenstates with the eigenspaces of the measurement operator $\tilde{M}$. Specifically, in the case where the spectrum of M takes the form $\{-\|M\|, \ldots, \|M\|\}$, the bound is saturated when one of the eigenstates is contained in the $+\|M\|$-eigenspace of $\tilde{M}$ and the other one is contained in the $-\|M\|$-eigenspace of $\tilde{M}$. On the other hand, for operators with spectra of the form $\{0, \ldots, \|M\|\}$ (e.g., basis-state projectors), the expression in Eq. (49) is maximized when one eigenstate is fully contained in the 0-eigenspace and one is fully contained in the $+\|M\|$-eigenspace. Note that, in this case, the following tighter bound holds:

$$\Delta \leq \|M\| \ \||\mathbf{z}\rangle\langle\mathbf{z}| - |\mathbf{z}'\rangle\langle\mathbf{z}'|\|_{\mathrm{Tr}}, \tag{52}$$

as used in Appendix A.

https://doi.org/10.52202/079017-1004

D Universality of weight-tied quantum models

Theorem 3 (Universality of weight-tied quantum models, in analogy to Theorem 3 in Bai et al. (2019))**.** *Let $\mathcal{E}_i(\cdot) = U(\theta^{(i)})(\cdot)U^\dagger(\theta^{(i)})$ be a channel corresponding to the PQC at depth i. Additionally, we define an map $R' : \mathbb{C}^{2^Q \times 2^Q} \to \mathbb{R}^K$ that describes performing measurements of expectation values with respect to an ensemble of K observables and storing the respective outcomes in a K-dimensional vector. Typically, we string this together with an upsampling map $\mathcal{I}_u$ so that $R = \mathcal{I}_u \circ R'$ maps to $\mathbb{R}^n$. Let $S_{\mathbf{z}}$ be a unitary encoding that encodes a vector from $\mathbb{R}^n$ into a quantum state in $\mathbb{C}^{2^Q}$, $S_{\mathbf{z}} : |0\rangle \mapsto |\mathbf{z}\rangle$. This allows us to define an encoding map $\mathcal{S}$ that maps $\mathbf{z}$ to a density matrix, so that $\mathbf{z} \mapsto |\mathbf{z}\rangle\langle\mathbf{z}|$, where the evolution of hidden layers can then be written as*

$$\mathbf{z}^{(i+1)} = R(\mathcal{E}_i \circ \mathcal{S}\mathbf{z}^{(i)}), \quad 0 \leq i < L, \quad \mathbf{z}^{(0)} = \mathbf{x}. \tag{53}$$

Then, a sequence of such layers can be replicated exactly by an input-injected, weight-tied network. Specifically,

$$\widetilde{\mathbf{z}}^{(i+1)} = R_L(E_z\widetilde{\mathbf{z}}^{(i)} + E_x\mathbf{x}) \tag{54}$$

where

$$R_L = \begin{bmatrix} R \\ \vdots \\ R \end{bmatrix}, \quad E_z = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ \mathcal{E}_1 \circ \mathcal{S} & 0 & 0 & \cdots & 0 \\ 0 & \mathcal{E}_2 \circ \mathcal{S} & 0 & \cdots & 0 \\ & & \ddots & & 0 \\ 0 & 0 & 0 & \mathcal{E}_{L-1} \circ \mathcal{S} & 0 \end{bmatrix}, \quad E_x = \begin{bmatrix} \mathcal{E}_0 \circ \mathcal{S} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{55}$$

yields after L iterations an output

$$\widetilde{\mathbf{z}}^{(L)} = \begin{bmatrix} \mathbf{x} \\ \mathbf{z}^{(1)} \\ \vdots \\ \mathbf{z}^{(L)} \end{bmatrix} \tag{56}$$

containing the output $\mathbf{z}^{(L)}$ of the non weight-tied network. This follows by construction, similarly to Theorem 3 in Bai et al. (2019).

E Experiments – Hyperparameter search

We used the validation set to search over hyperparameters listed in Table 5. For each hyperparameter, we note the ranges we considered, as well as the optimal value for each dataset.

Table 5: Optimal hyperparameters for each model. The search space we considered was: learning rate $\{0.005, 0.0075, 0.01, 0.05, 0.1\}$, number of warm-up steps $\{1875, 2355, 3750\}$, number of warm-up layers $\{1, 2\}$, weight of the Jacobian loss $\{0, 0.5, 0.8\}$, frequency of the Jacobian loss $\{0.0, 0.5, 0.8, 1.0\}$.

		IMPLICIT+WARMUP		
Hyperparamter	MNIST-4	MNIST-10	FashionMNIST-10	CIFAR-10
Learning rate	0.05	0.05	0.05	0.01
Num. warm-up steps	1875	1875	1875	2350
Warm-up layer depth	1	1	1	1
Jac. loss weight	0.0	0.8	0.8	0.8
Jac. loss freq.	0.0	1.0	0.8	1
		IMPLICIT		
Hyperparamter	MNIST-4	MNIST-10	FashionMNIST-10	CIFAR-10
Learning rate	0.05	0.05	0.05	0.05
Jac. loss weight	0.0	0.8	0.5	0.0
Jac. loss freq.	0.0	1.0	0.8	0.0
		DIRECT - 10 layers		
Hyperparamter	MNIST-4	MNIST-10	FashionMNIST-10	CIFAR-10
Learning rate	0.1	0.05	0.05	0.0075
		DIRECT - 5 layers		
Hyperparamter	MNIST-4	MNIST-10	FashionMNIST-10	CIFAR-10
Learning rate	0.05	0.05	0.0075	0.0075
		DIRECT - 2 layers		
Hyperparamter	MNIST-4	MNIST-10	FashionMNIST-10	CIFAR-10
Learning rate	0.05	0.05	0.05	0.01
		DIRECT - 1 layer		
Hyperparamter	MNIST-4	MNIST-10	FashionMNIST-10	CIFAR-10
Learning rate	0.05	0.05	0.01	0.01

https://doi.org/10.52202/079017-1004

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims of the abstract reflect the findings of the paper to the best of our judgement.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Information regarding that can be found in the Results and Conclusion and Open Problems sections.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Our main theoretical results are in Theorem 3 and Appendices A and B. To the best of our judgement, our Theorem provides a full set of assumptions in combination with Definition 1. Furthermore, we openly discuss the limitations of our arguments in the appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Our Section 3 contains information regarding all our models and baseline models we are referring to. With a full publication, we will publicly share explicit code as well as access to weights and biases of our experiments to ensure full reproducibility.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

 https://doi.org/10.52202/079017-1004

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   All datasets used in the paper are publicly available. The code is publicly available on GitHub at the following link: `https://github.com/martaskrt/qdeq`

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The main hyperparameters and methods relevant to reproducing the proposed method are reported. Additional hyperparameters will be supplied with publication of the code.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper reports individual runs, and has been phrased so as to not make claims about statistical significance or guarantees of broader applicability.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: For all experiments, the main computational hardware and the runtime is reported. When choosing the level of detail of this reporting, the independence of the paper's claims to the scaling of these resources was considered.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors have read the Code of Ethics, and report that the work here is in compliance.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

https://doi.org/10.52202/079017-1004

Justification: The paper proposes a change in methodology in the already well-established field of image classification. Furthermore, the limited scale of current quantum computers reduces the chance of immediate impact, also within this field.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The work in the paper does not pose a risk of misuse, and no new datasets are released as part of the paper.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets and code used is cited, and the relevant licenses named.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper uses existing datasets, and does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No research involving crowdsourcing or human subjects is included in the paper.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No research involvingcrowdsourcing or human subjects is included in the paper.

https://doi.org/10.52202/079017-1004

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.