Identifying Phases with Quantum Convolutional Neural Network

Zhangjie Qin, Ho Lun Tang, Louie Hong Yao

Virginia Tech

Abstract: In this project, we explore the application of a quantum convolutional neural network model for phase identification. Our investigation involves training regression and classification models to identify phases and phase transitions within both the Transverse Field Ising Model and $\mathbb{Z}_2 \times \mathbb{Z}_2$ Symmetry Protected Topological Phase model. While our regression model, utilizing shallow circuitry, performs reasonably, the classification models initially underperform. However, leveraging GPU acceleration, we enhance the classification models by increasing circuit depth, resulting in significant performance improvements. Additionally, we evaluate the robustness of our models on IonQ's quantum device using the AWS Bracket hybrid job tool.

Contents

1 Introduction

The study of phase transitions holds much importance in various areas, particularly in condensed matter physics and statistical mechanics. These transitions represent fundamental shifts in the collective behavior of systems, unveiling critical insights into their underlying properties and dynamics. Understanding phase transitions enables us to elucidate the macroscopic behavior emerging from microscopic interactions, paving the way for advancements in materials science, quantum mechanics, and thermodynamics. Moreover, phase transitions often underpin the emergence of novel phenomena such as superconductivity, magnetism, and critical phenomena, which have far-reaching implications across multiple fields, including technology, engineering, and even cosmology.

Inspired by the remarkable effectiveness of Convolutional Neural Networks (CNN) in machine learning, Quantum Convolutional Neural Networks (QCNN) [1] have emerged as a highly promising approach within the realm of quantum computing. QCNN aims to leverage the intrinsic power of quantum computing and exhibits adaptability for implementation on quantum devices. Another advantage of QCNN, as highlighted in previous

studies, is its absence of barren plateaus [2]—a longstanding challenge for variational quantum algorithms. Both of these advantages make QCNN a tangible solution for addressing challenges related to phase identification [3].

In this project, we utilize QCNN to identify phases for two distinct models: the transverse field Ising model and the symmetry-protected topological system. Through classical simulation, we achieve satisfactory results, with prediction accuracy exceeding 90% for regression models with shallow circuit depth. Furthermore, we assess the robustness of our regression models on noisy datasets and on the IonQ Aria device on the AWS platform. Remarkably, the regression models remain effective even on noisy datasets, showcasing their resilience to noise. However, when deployed on the real device with circuit noise, while the regression model retains some prediction power for the transverse field Ising model (TFIM), it fails to accurately predict SPT phases due to the noise. To gain further insight into how the regression model captures order parameters, we evaluate the models on random 8-qubit states and compare the predictions with ground truth. Surprisingly, instead of learning to capture information about order parameters for general states, the regression models focus solely on capturing useful information for ground states.

The initial version of the classification models, with shallow circuit depths, exhibited poor performance. However, leveraging Nvidia GPU capabilities, we extended our circuits four times deeper, enabling greater model expressibility. The deep circuit versions of the classification models demonstrated significant improvements, particularly in the case of the transverse field Ising model (TFIM). They successfully captured the sharp change near the critical point indicative of the phase transition. Similarly, for the symmetry-protected topological (SPT) phases, the model showed improvement, although it still grappled with the inherent challenge of predicting two phase transitions simultaneously.

This report is structured as follows: in the subsequent section, we introduce the two models we focus on, namely the transverse field Ising model (TFIM) and the $Z_2 \times Z_2$ symmetry-protected topological (SPT) phase. We present our methods and network structures in section 3, followed by an analysis of the model performance in section 4. We conclude with a brief summary of this project in section 5. In the appendix A, we review the Nvidia GPU speedup of quantum circuit simulations and benchmark the speed using our QCNN structures on the Nvidia A100 GPU.

2 Background

We apply the QCNN to two physics systems: the transverse field Ising model and the symmetry-protected topological phase model. In this section, we provide a concise review of both models.

2.1 Transverse Field Ising Model

The 1-dimensional transverse field Ising model (TFIM) contains spin with nearest-neighbor interaction. The Hamiltonian reads as

$$H = -J \sum_{\langle i,j \rangle} Z_i Z_j - h \sum_i X_i, \tag{2.1}$$

where X, Z are Pauli matrices. J is the Ising interaction between the nearest neighbor and h is the external field.

Though simple in structure, the Transverse Field Ising Model (TFIM) embodies a continuous phase transition, making it an excellent illustrative example for fundamental concepts. In the thermodynamic limit, the phase transition manifests when $h = |J|$. Below this threshold, an ordered phase emerges, with an antiferromagnetic order for $J < 0$ and a ferromagnetic order for $J > 0$. Conversely, beyond $h > |J|$, the system transitions into the paramagnetic order, where all spins align with the external field. In this work, we will focus on the ferromagnetic TFIM. The phase transition can be observed through the magnetization in z-direction m_z. $|m_z| > 0$ in the ferromagnetic order as the spins align in the z-direction. In the paramagnetic phase, the spins align in the x-direction, thus $m_z = 0$. In this work, we study this model with 1D chain geometry and open-boundary condition.

2.2 Symmetry Protected Topological Phase

The Hamiltonian reads as

$$H = -J \sum_{i=1}^{N-2} Z_i X_{i+1} Z_{i+2} - h_1 \sum_{i}^{N} X_i - h_2 \sum_{i=1}^{N-1} X_i X_{i+1} \tag{2.2}$$

In the Hamiltonian, X, Y, and Z represent Pauli matrices. When $h_1 = h_2 = 0$, the ground state of the Hamiltonian forms the linear cluster state. This assertion arises because the term $Z_i X_{i+1} Z_{i+2}$ precisely acts as the stabilizer for the linear cluster state. For open boundary conditions, the left and right boundary stabilizers are given by $Z_1 X_2 (X_{N-1} Z_N)$. The linear cluster state embodies a one-dimensional symmetry-protected topological state with $\mathbb{Z}_2 \times \mathbb{Z}_2$ symmetry. The parity symmetry within the linear cluster state can be generated by the product of stabilizers, explicitly denoted as $P_{\text{even(odd)}} = \prod_{i \in \text{even(odd)}} X_i$, representing on-site and non-local string symmetries. The identification of the symmetry-protected phase can be accomplished via the string order parameter. For instance, the string order parameter $S = Z_i X_{i+1} .. X_{j-1} Z_j$ of the ideal linear cluster state equals 1 (trivial phase is 0). When $h_2 = 0$, the Hamiltonian is exactly solvable using the Jordan-Wigner transformation. The phase diagram of the system is depicted in Fig.1 [1]. In this project, for simplicity, we set $h_1 = 0$ and denote $h_2 = h$.

3 Method and Network Sturcture

3.1 Data Generation

We use the exact diagonalization method to generate 50,000 data points, comprising ground states, eigenvectors, and magnetization or string order parameters for 8-qubit systems.

In our TFIM sampling, we compute the sample magnetization density, as depicted in Fig.2(a). When h/J is small, the Z_2 symmetry breaks, resulting in two degenerate ground states within the ordered phase, characterized by $|\langle M \rangle| > 0$. As h/J increases, the ground state's magnetization density approaches 0, and the Z_2 symmetry is restored. Notably, in the 8-qubit system, the critical point occurs around 0.02, primarily due to significant

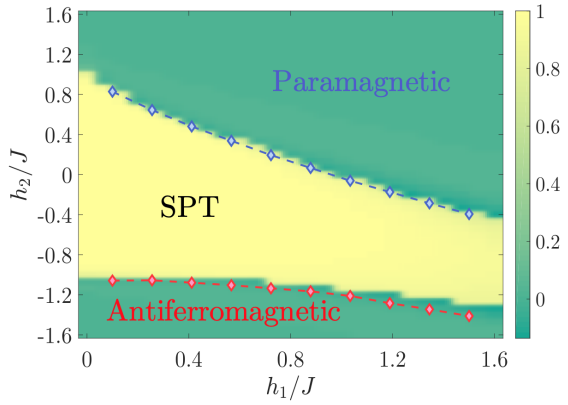

Figure 1. The phase diagram of SPT model with different parameters [1]

finite-size effects. In the SPT sampling, we compute the expectation of the string order parameter $S = Z_0X_1X_3X_5Z_6$, plotted in Fig.2(b). The SPT phase is distinctly identified by $\langle S\rangle > 0$. However, due to finite-size effects, it doesn't reach 1.

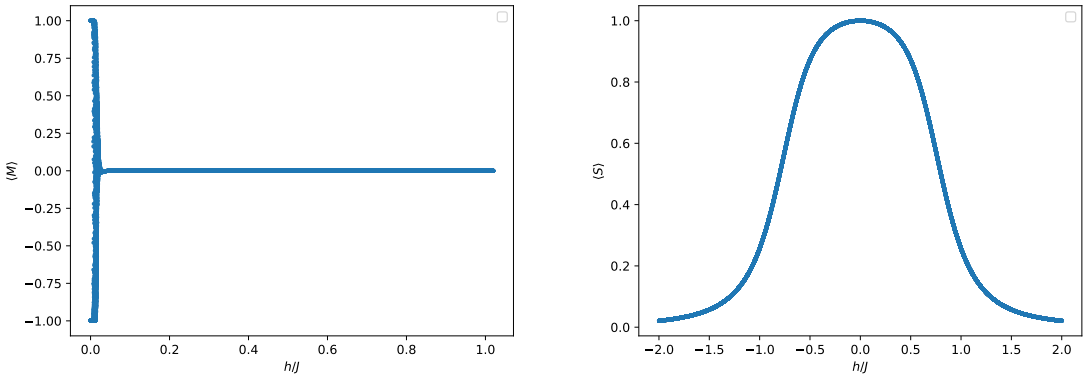

Figure 2. (a) The magnetization density as a function of the ratio h/J from the sampled data, with $L = 8$. Due to the extremely small system size and thus the strong finite-size effect, the critical point is shifted to around 0.02. (b) The string order parameter as a function of the ratio h/J from the sampled data, with $L = 8$. Due to the extremely small system size and thus the strong finite-size effect, $\langle S\rangle$ doesn't research to 1.

3.2 Quantum Circuit and Training

We adapt the circuit ansatz and network structure from [4]. In our QCNN training, we use the $SU4$ circuit ansatz as plotted in Fig.3(a). Our network consists of three convolutional layers, where the i-th layer convolutes the k-th qubit with the $(k + i + 1)$-th qubit for $k = 0, 1, ..., 7$. Following each convolutional layer, we apply a pooling layer as plotted in Fig.3(b). In the end, we only measure the fourth qubit as the output of this network.

	Models	Accuracy
Classification Models	TFIM	0.86
	SPT	0.78
Regression Models	TFIM	0.99
	SPT	0.92

Table 1. The testing accuracy: For the regression model, the accuracy is calculated based on predicting the ordered phase when the predicted order parameter is greater than 0.05.

In this project, we explore two distinct models: the regression model and the classification model. In the regression model, we gauge the expectation value of the Pauli Z operator as the output and contrast it with the expectation value of the order parameter $\langle M \rangle$ or $\langle S \rangle$. We employ mean squared error (MSE) as the cost function for evaluation. In the classification model, we aim to predict the phase label. However, precise identification of the phases is challenging due to extreme finite-size effects. To simplify, we define the ordered phase (or SPT) as when $\langle M \rangle > 0.05$ (or $\langle S \rangle > 0.05$). We compute the probability occurrence of the 4-th qubit and employ cross-entropy as the cost function for assessment.

4 Result and Analysis

4.1 Shallow Circuits

We train the regression model (and the classification model) for 100 (200) episodes, respectively, with a batch size of 25. The parametrized quantum circuit is optimized using the Adam optimizer provided by the PennyLane package on the CPU. The training curve is plotted in Fig.(5). Throughout the training process, we split the dataset into a 90% training set and a 10% testing set. The accuracy on the testing set is presented in Table(4.1). As observed from the table, the regression model outperforms the classification model significantly, indicating that the parametrized quantum circuit excels in predicting continuous variables.

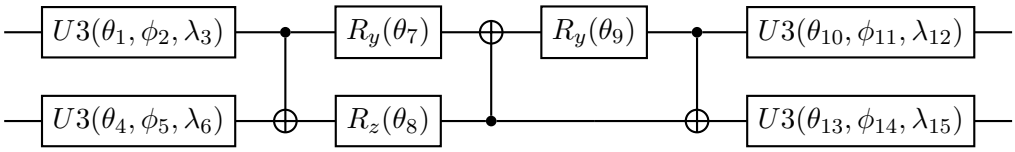

Figure 3. QCNN convolutional circuit ansatz

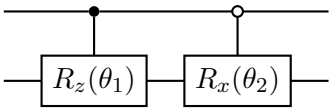

Figure 4. Pooling layer ansatz.

To comprehend the behavior of the trained QCNN, we calculate the ground state and the expectation value of the order parameters for 1000 points and compare the predictions from the regression model with the exact results, which are plotted in Fig.(6). From the

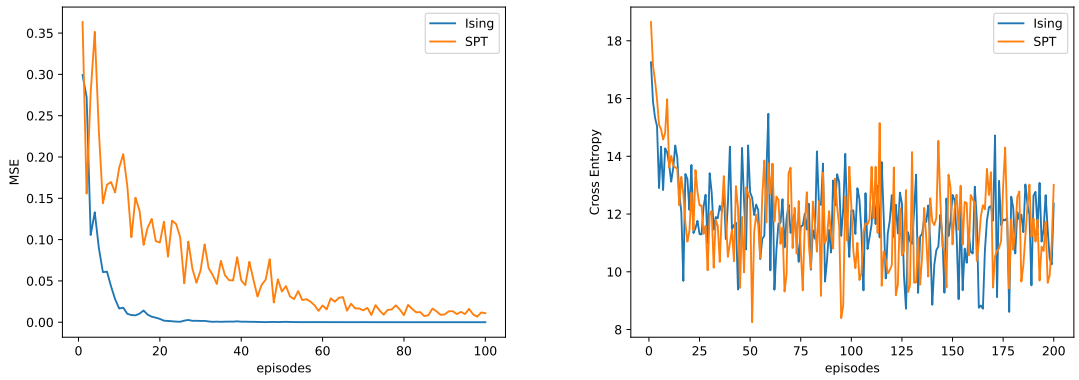

Figure 5. The training curve of the regression model and the classification model.

figures, it's evident that the regression QCNN predicts the TFIM very accurately. However, for the SPT, while the regression model captures the trend of the order parameter well, it doesn't precisely match the values.

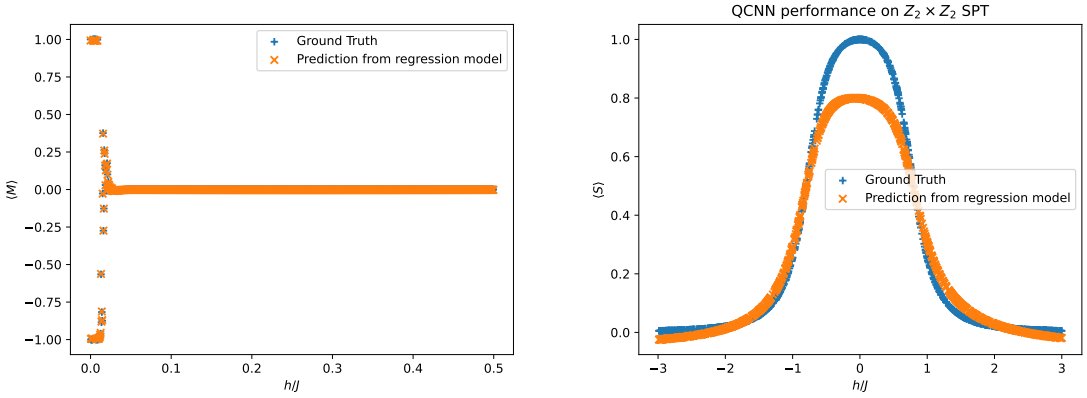

Figure 6. Prediction from regression model vs Exact

For the classification model, we compute the probability of predicting the ordered phase for the same 1000 points and present the results in Fig.(7). The TFIM prediction reveals that the classification model struggles to accurately classify the degenerate ordered phase. This challenge stems from the distant nature of the degenerate ground states within the ordered phase, which belong to the same phase but are challenging for the shallow QCNN to distinguish simultaneously. Regarding SPT classification, the model appears to grasp the underlying pattern. However, due to expressibility constraints, the model's predictions may not be entirely accurate. This limitation arises because states far from each other could potentially belong to the trivial phase, as observed in the phase diagram (Fig.1), where the trivial phase exists on both the left and right sides of the SPT phase. Furthermore, special

states like Symmetry Protected Topological (SPT) states only occupy a small portion of the Hilbert space, further complicating the model's ability to differentiate them effectively.

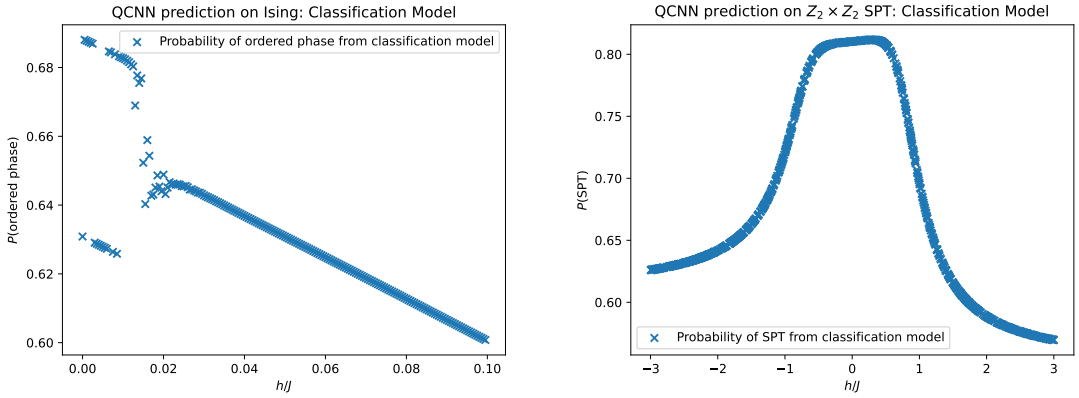

Figure 7. Probability of predicting the ordered phase

4.1.1 States with Correlated Two-Qubit Noises

We proceed to assess the robustness of our regression models against correlated noise. Specifically, we introduce one layer of two-qubit correlated noises, namely $e^{i\alpha X_i X_j}$ and $e^{i\alpha Z_i Z_j}$, to the states in our testing set. Here, α follows a uniform distribution within the range $[-\pi/10, \pi/10)$, and i, j are randomly selected from 8 qubits. We then plot the predictions versus the real expectation value of the order parameters before the introduction of noise, as depicted in Fig. 8. For both the Ising model and the SPT phases, although the predictions are influenced by the correlated noise, they still capture the trend of the change and effectively predict the phase transitions.

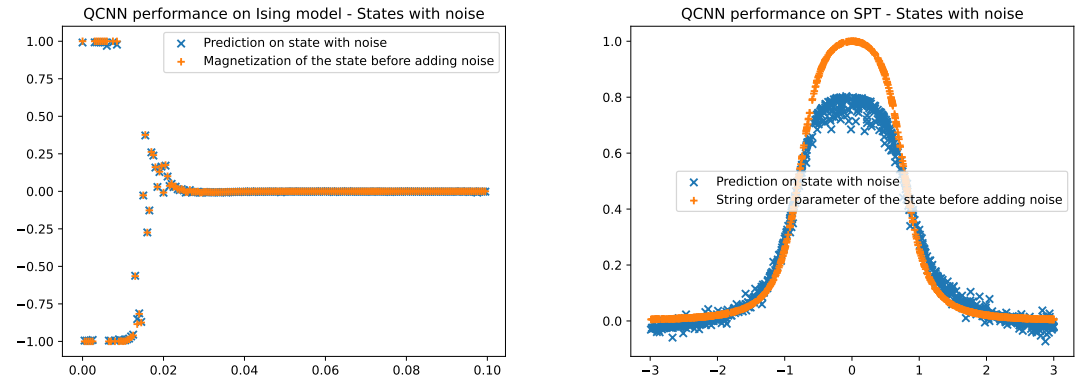

Figure 8. Performance of the regression model on states with noises.

4.1.2 Performance on quantum simulator and Real Quantum device

We also conducted tests on our model using the simulator and IonQ Aria-1 device. AWS Bracket provides a tool for running hybrid circuits on classical AWS resources and QPUs, enabling convenient testing of our models. However, due to the limitations of the current device, training the circuit directly on the device is challenging. Instead, we opted to train our model classically and evaluated how noises in the quantum circuit would affect the robustness of prediction in the model. Given that shallow circuit classification models underperformed even in the absence of noise, our attention shifted exclusively toward evaluating the regression model. We successfully reconstructed and deployed the quantum convolutional neural network, incorporating the parameters derived from the classical training part, on both a simulator and an actual quantum device. And we tested the prediction for both the Transverse Field Ising Model and the Symmetry-Protected Topological Phase Model, specifically within the 8-qubit example. The results were obtained from the expectation value of 1,000 measurements on real quantum devices, depicted in Fig. 9, indicate that for the Ising model, despite circuit noises causing deviations in predictions from exact values, a discernible trend of phase transitions is still evident as magnetization decreases nearly to zero with increasing h. However, due to the dominant crosstalk errors inherent in large-size trapped ion systems, the entanglement characteristic of SPT phases is disrupted, leading to a complete loss of predictive power for regression models on SPT phases.

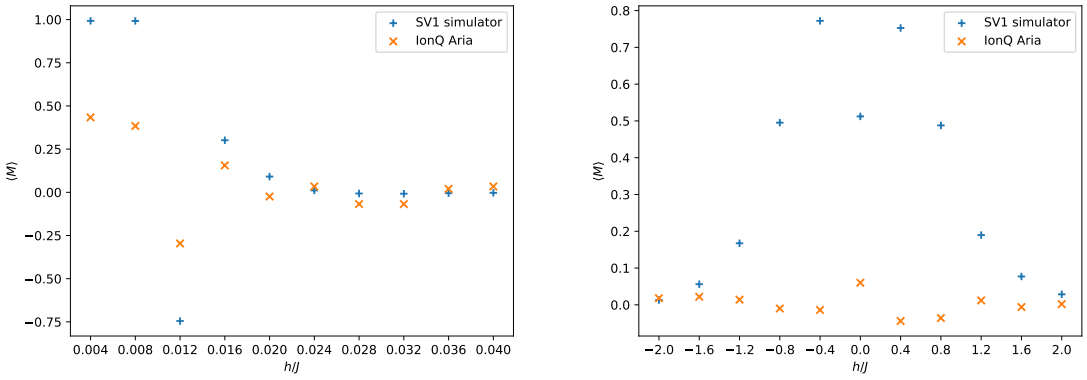

Figure 9. Prediction from IonQ Aria-1 vs prediction from AWS SV1 simulator

4.1.3 Prediction Power on a General State

Thus far, our predictions have primarily focused on ground states and those closely resembling them. However, the regression model aims to convey nonlocal information to a single qubit. We have a clear understanding of the specific information we seek—total magnetization for the Ising model and the string order parameter for the SPT phase. To gauge the model's ability to acquire this information for general states, beyond those near ground

states, we undertake a new test. We compute magnetization and string order parameters of randomly generated 8-qubit states using the Haar measure. Subsequently, we utilize the regression models to predict on these states and compare the results against exact values. The findings, illustrated in Fig.10, reveal that the models struggle to accurately capture the expectation value of order parameters for general states. Contrary to expectations of a broad approach to information propagation, the models seem to have specialized in capturing nonlocal information tailored specifically to ground states.

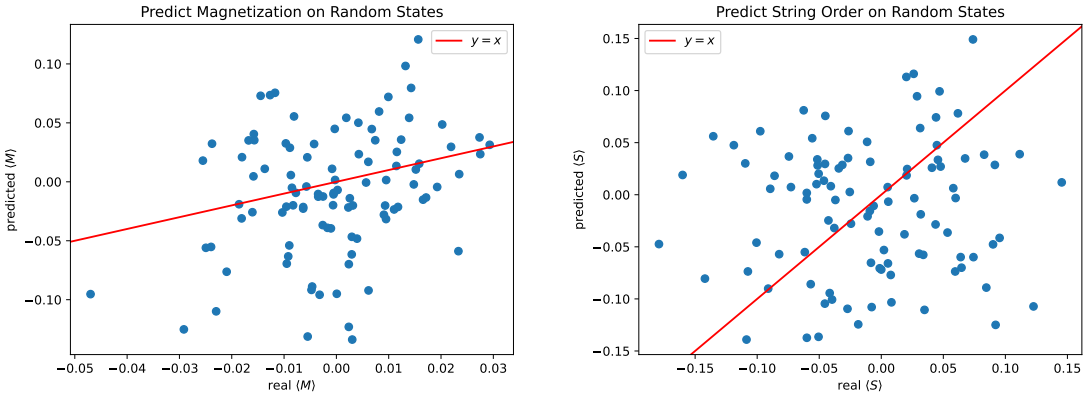

Figure 10. Performace of regression models on random states. The red line is a reference for good predictions.

4.2 Deep Circuits

To enhance the performance of the classification model, we extended the expressibility by employing deeper circuits. With the assistance of Nvidia GPU and the PennyLane-lightning package, we were able to expand our circuits to four times their original depth. A brief introduction to GPU acceleration and a benchmarking of speed are provided in Appendix A.

In this section, we assess the performance of classification models with deeper circuits. Fig. 11 showcases the probability of predicting the ordered (or SPT) phase across varying h/J values using ground states. The figure reveals that although the classification model for the Ising model isn't yet perfect and encounters challenges near the critical point, it discerns a significant change near the critical value $h/J \approx 0.02$. Consequently, it accurately classifies both distinct ground states in the ordered phase and the ground states in the disordered phase away from the critical point. This result differs from the shallow circuit scenario, where the model compromises its ability to correctly classify the disordered phase to accommodate both ground states of the ordered phase. Moreover, due to the expanded parameter space, the expressibility of the SPT classification model also improves. Despite still facing challenges, the model can now predict the left tail accurately. Unlike the Ising model, here, we predict two different phase transitions simultaneously, which adds

complexity. Nonetheless, the performance of this deeper model suggests the classification model's potential to predict phases correctly when the circuit depth is sufficient.

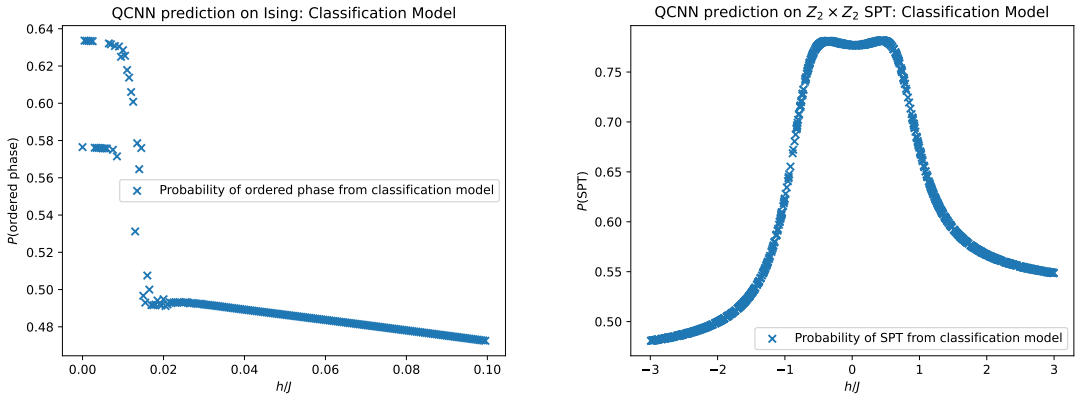

Figure 11. Performace of classification models with deep circuit depth

5 Summary

In this project, we utilized quantum convolutional neural networks to tackle the identification of phases and phase transitions within physical systems. Through training regression and classification models, we aimed to predict order parameter expectations and phase labels for both the transverse field Ising model and the $\mathbb{Z}_2 \times \mathbb{Z}_2$ symmetry-protected topological system.

Our regression models performed remarkably well, even with shallow circuit depths. We tested their robustness on noisy datasets and IonQ's real device. On the noisy dataset, the models perform reasonably well. On the real device, while the Ising model predictions remained relatively accurate despite the noise, our regression model encountered challenges in predicting SPT phases due to crosstalk noise disrupting entanglement structures. Additionally, testing on random states away from ground states revealed limited predictive power, suggesting that our regression models specialize in capturing information tailored to ground states.

Initially, our classification models with shallow circuit depths demonstrated poor performance. However, upon increasing circuit depth, we observed improvements in model expressibility. The classification model for the Ising model began capturing changes near the critical point, although inaccuracies persisted near the critical point. Meanwhile, the SPT classification model exhibited reasonable accuracy in predicting one of the tails. Despite the inherent difficulty of predicting two phase transitions simultaneously, deeper circuit depths hold promise for enhancing the classification model's accuracy further.

Acknowledgments

The authors extend their sincere gratitude to the sponsors and organizers for hosting a successful and enjoyable QHack event. Furthermore, the authors are deeply grateful to NVIDIA, Denvr Dataworks, and AWS for their generous support, which made this project possible. Special thanks to ChatGPT for its assistance in writing this report.

A GPU Speed-up with Pennylane and Nvidia cuQuantum

Quantum computing holds immense promise for solving complex problems and demonstrating quantum advantages, especially within quantum physics-related domains. However, the majority of current devices are constrained by their small size and high levels of noise, limiting their ability to effectively support research on quantum algorithms. To showcase the capabilities of algorithms in the current situation, large-scale classical simulations of quantum computing models are necessary. GPUs, with their unique properties, offer numerous opportunities to accelerate these simulations and support research in quantum computation and quantum algorithms. The software cuQuantum [5], developed by NVIDIA, enables researchers to efficiently simulate their quantum circuits through distributed computation on multiple GPUs. Together with the PennyLane SDK, quantum simulations, Variational Quantum Algorithms, and Quantum Machine Learning can be efficiently and easily run on GPUs, reaching quantum volumes that CPUs cannot access. This advancement facilitates the exploration of quantum computing potential and fosters advancements in quantum algorithm research.

Here, we benchmark the speed of QCNN operations on a 2019 MacBook Pro with Quad-Core Intel Core i5 and on a single Nvidia A100 GPU using the PennyLane simulator. The results are depicted in Fig. 12. A system size of more than 22 qubits is already inaccessible within reasonable time and memory constraints on the CPU. However, the Nvidia A100 GPU extends the limit to 30 qubits. Additionally, we benchmark the runtime of different QCNN layers on a 16-qubit system on both CPU and GPU. The results demonstrate that the GPU significantly reduces the runtime for multi-layer QCNN operations.

References

[1] I. Cong, S. Choi and M.D. Lukin, *Quantum convolutional neural networks*, *Nature Physics* **15** (2019) 1273–1278.

[2] A. Pesah, M. Cerezo, S. Wang, T. Volkoff, A.T. Sornborger and P.J. Coles, *Absence of barren plateaus in quantum convolutional neural networks*, *Physical Review X* **11** (2021) .

[3] I. MacCormack, C. Delaney, A. Galda, N. Aggarwal and P. Narang, *Branching quantum convolutional neural networks*, 2020.

[4] T. Hur, L. Kim and D.K. Park, *Quantum convolutional neural network for classical data classification*, *Quantum Machine Intelligence* **4** (2022) 3.

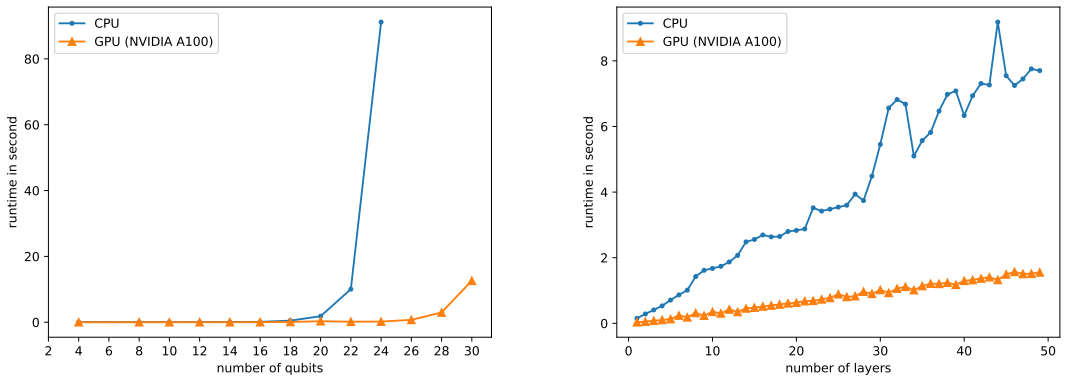

Figure 12. Performance of QCNN operations on CPU and GPU: (a) We benchmark one layer of QCNN operation on different numbers of qubits. (b) We measure the runtime of QCNN operation on 16 qubits with different numbers of layers.

[5] H. Bayraktar, A. Charara, D. Clark, S. Cohen, T. Costa, Y.-L.L. Fang et al., *cuquantum sdk: a high-performance library for accelerating quantum science*, in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 1, pp. 1050–1061, IEEE, 2023.