Error-tolerant quantum convolutional neural networks for symmetry-protected topological phases

Petr Zapletal [1,2,*] Nathan A. McMahon [1] and Michael J. Hartmann [1]

[1]*Department of Physics, Friedrich-Alexander University Erlangen-Nürnberg (FAU), Erlangen, Germany*
[2]*Department of Physics, University of Basel, Klingelbergstrasse 82, 4056 Basel, Switzerland*

The analysis of noisy quantum states prepared on current quantum computers is getting beyond the capabilities of classical computing. Quantum neural networks based on parametrized quantum circuits, measurements and feed-forward can process large amounts of quantum data to reduce measurement and computational costs of detecting nonlocal quantum correlations. The tolerance of errors due to decoherence and gate infidelities is a key requirement for the application of quantum neural networks on near-term quantum computers. Here we construct quantum convolutional neural networks (QCNNs) that can, in the presence of incoherent errors, recognize different symmetry-protected topological phases of generalized cluster-Ising Hamiltonians from one another as well as from topologically trivial phases. Using matrix product state simulations, we show that the QCNN output is robust against symmetry-breaking errors below a threshold error probability and against symmetry-preserving errors provided the error channel is invertible. This is in contrast to string order parameters and the output of previously designed QCNNs, which vanish in the presence of any symmetry-breaking errors. To facilitate the implementation of the QCNNs on near-term quantum computers, the QCNN circuits can be shortened from logarithmic to constant depth in system size by performing a large part of the computation in classical postprocessing. These constant-depth QCNNs reduce sample complexity exponentially with system size in comparison to the direct sampling using local Pauli measurements.

I. INTRODUCTION

Existing noisy intermediate-scale quantum (NISQ) computers can perform computations that are challenging for classical computers [1]. However, quantum computing hardware and quantum algorithms need to be further developed to enable the exploitation of quantum computers in areas such as the simulation of many-body systems [2,3] and machine learning [4]. One of the major challenges in developing scalable quantum computers is the characterization of noisy quantum data produced by near-term quantum hardware. With increasing system size, standard characterization techniques using direct measurements and classical postprocessing become prohibitively demanding due to large measurement counts and computational efforts. While many local properties can be efficiently determined using randomized measurements [5], global properties of quantum states are typically hard to estimate.

Quantum machine learning techniques based on the direct processing of quantum data on quantum processors can substantially reduce the measurement costs, including quantum principle component analysis [6], quantum autoencoders [7–9], certification of Hamiltonian dynamics [10,11], quantum reservoir processing [12]. Moreover, quantum neural networks based on parametrized quantum circuits, measurements and feed-forward can process large amounts of quantum data, to detect nonlocal quantum correlations with reduced measurement and computational efforts compared to standard characterization techniques [13–17]. A key requirement for employing quantum neural networks to characterize noisy quantum data produced by near-term quantum hardware is the tolerance to errors due to decoherence and gate infidelities.

The characterization of nonlocal correlations in quantum states is of key importance to condensed matter physics. It is required for the classification of topological quantum phases of matter [18–21] and for understanding new strongly correlated materials [22] such as high-temperature superconductors [23]. Classical machine learning tools for the recognition of topological phases of matter have recently been studied, uncovering phase diagrams from data produced by numerical simulations [24–26] and measured in experiments [27–30]. Moreover, quantum many-body states belonging to topological quantum phases have been prepared on quantum computers using exact matrix product state representations [31], unitary quantum circuits [32], and measurement and feed-forward [33]. Properties of topological phases have been probed on quantum computers by measuring characteristic quantities [31,34] such as string order parameters (SOPs) [35,36]. The detection of topological phases can be enhanced via the processing of measurement data on a classical computer [37]. Classical machine learning algorithms have been shown to classify topological quantum phases from

*Contact author: petr.zapletal@fau.de

classical shadows formed by randomized measurements [38]. However, the rapidly increasing sample complexity with system size remains an outstanding problem for such approaches.

In Ref. [14], quantum convolutional neural networks (QC-NNs) have been proposed to recognize symmetry-protected topological (SPT) phases [18,19] with reduced sample complexity compared to the direct measurement of SOPs. Such QCNNs can be trained to identify characteristics of SPT phases from training data [14,39,40]. Alternatively, QCNNs can be analytically constructed to mimic renormalization-group flow [14,41], a method for classifying quantum phases [22]. A shallow QCNN has been implemented on a 7-qubit superconducting quantum processor in Ref. [42]. This QCNN has exhibited robustness against incoherent errors on the NISQ device, which allowed for the recognition of a SPT phase with a higher fidelity than the direct measurement of SOPs. However, the propagation of errors leads to a rapid growth of error density in deeper QCNNs due to the reduction of qubit number from one QCNN layer to the next, which represents a central problem.

Here we overcome this problem by designing QCNNs for generalized cluster-Ising models that can tolerate incoherent errors. The QCNN circuits are constructed by alternating layers, which mimic renormalization-group flow, and new layers, which correct incoherent errors. Due to the tolerance to errors, the QCNNs recognize SPT phases of exact ground states provided access to only noisy states, which approximate the former on NISQ devices. Apart from distinguishing SPT phases from topologically trivial phases as previously shown in Refs. [14,40–42], we demonstrate that QCNNs constructed here can recognize two SPT phases from one another.

Using matrix product state (MPS) simulations, we show that the QCNN output is robust against symmetry-breaking errors below a threshold error probability. This enables new quantum phase recognition capabilities for QCNNs in scenarios where SOPs and previous QCNN designs [14] are impractical. SOPs rapidly vanish with an increasing length for any probability of symmetry-breaking errors [43], whereas the QCNN proposed in Ref. [14] rapidly concentrates symmetry-breaking errors with increasing depth leading to a vanishing output for any error probability.

In addition to the tolerance to symmetry-breaking errors, the QCNNs constructed here tolerate symmetry-preserving errors if the error channel is invertible. The error tolerance is limited close to phase boundaries as the threshold error probability decreases with diverging correlation lengths. Nonetheless, a sharp change in the QCNN output at the phase boundaries allows us to precisely determine critical values of Hamiltonian parameters.

To facilitate the implementation of QCNNs on near-term quantum computers, we show that the QCNN circuits constructed here can be shortened from logarithmic to constant depth in system size by efficiently performing a large part of the computation in classical postprocessing. The output of the QCNNs corresponds to the expectation value of a multiscale SOP, which is a sum of products of individual SOPs. The multiscale SOP can, in principle, be determined using direct Pauli measurements on the input state without using any quantum circuit. However, the constant-depth QCNN circuits we derive here reduce the sample complexity of measuring the

multiscale SOP exponentially with system size in comparison to direct Pauli measurements.

The remainder of this paper is structured as follows. In Sec. II, we introduce the generalized cluster-Ising model we consider before describing the construction of the QCNNs to analyze it in Sec. III. We investigate the robustness of the QCNN output against incoherent symmetry-preserving errors in Sec. IV and show how to design QCNNs that tolerate symmetry-breaking errors in Sec. V. We investigate the phase transition between two SPT phases in Sec. VI and study the tolerance to incoherent errors close to phase boundaries in Sec. VII. In Sec. VIII, we compare the sample complexity of QCNNs to the direct Pauli measurement of the input state before presenting concluding remarks and possible applications of error-tolerant QCNNs in Sec. IX.

II. GENERALIZED CLUSTER-ISING MODEL IN THE PRESENCE OF INCOHERENT ERRORS

We consider a one-dimensional chain of N qubits with open boundary conditions described by the generalized cluster-Ising Hamiltonian

$$H = -J_1 \sum_{j=2}^{N-1} C_j - J_2 \sum_{j=3}^{N-2} D_j$$

$$- h_1 \sum_{j=1}^{N} X_j - h_2 \sum_{j=1}^{N-1} X_j X_{j+1}, \qquad (1)$$

where $C_j = Z_{j-1} X_j Z_{j+1}$, $D_j = Z_{j-2} X_{j-1} X_j X_{j+1} Z_{j+2}$, and X_j as well as Z_j are Pauli operators on qubit j. The Hamiltonian exhibits $\mathbb{Z}_2 \times \mathbb{Z}_2$ symmetry generated by $P_{e/o} = \prod_{j=1}^{N/2} X_{2j/2j-1}$ as well as time-reversal symmetry (complex conjugation). The ground states $|\psi\rangle$ of the Hamiltonian belong to one of four phases: a paramagnetic phase, an antiferromagnetic phase, a ZXZ SPT phase and a $ZXXXZ$ SPT phase [44]. The ZXZ ($ZXXXZ$) SPT phase contains the ZXZ ($ZXXXZ$) cluster state, which is a stabilizer state with stabilizer elements C_j (D_j) and thus the ground state for $J_2 = h_1 = h_2 = 0$ ($J_1 = h_1 = h_2 = 0$). SPT phases are characterized by SOPs [35,36]. In particular, the SOPs

$$S_{jk} = Z_j \left(\prod_{i=1}^{(k-j)/2} X_{j+2i-1} \right) Z_k, \qquad (2)$$

$$T_{jk} = Z_j X_{j+1} Y_{j+2} \left(\prod_{i=2}^{(k-j)/2-2} X_{j+2i} \right) Y_{k-2} X_{k-1} Z_k, \qquad (3)$$

attain nonvanishing values in the ZXZ SPT phase and the $ZXXXZ$ SPT phase, respectively.

NISQ computers operate in the presence of noise due to decoherence and gate infidelities. To simulate errors that occur during the preparation of the many-body ground states $|\psi\rangle$ of the Hamiltonian (1) on NISQ devices, we consider an error channel

$$\rho = \mathcal{E}(|\psi\rangle\langle\psi|) = \sum_{l=1}^{4^N} K_l |\psi\rangle\langle\psi| K_l^\dagger, \qquad (4)$$
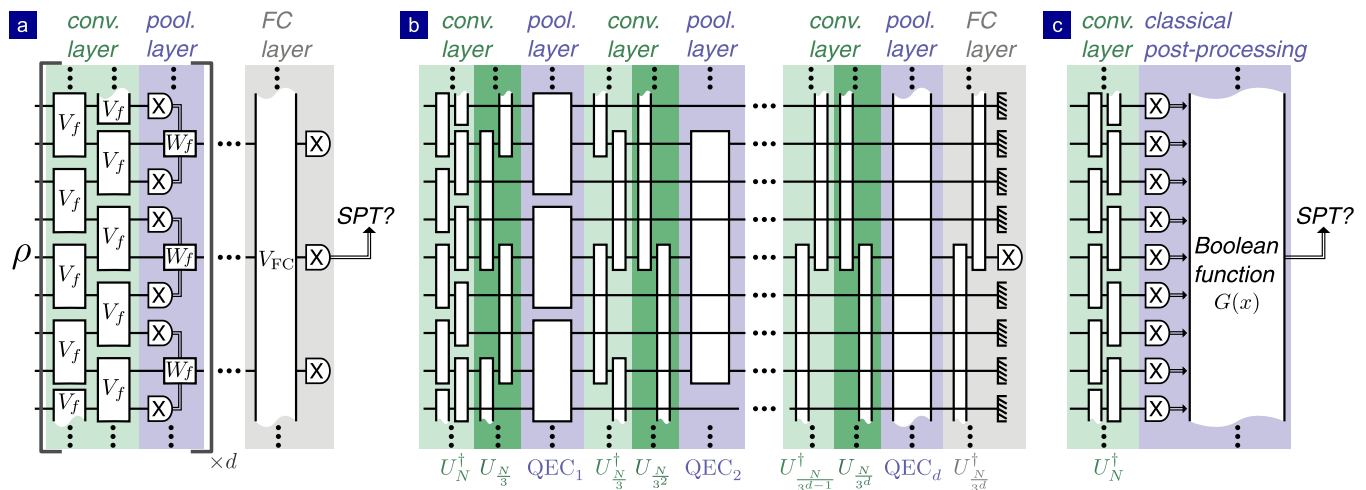
FIG. 1. Quantum convolutional neural network (QCNN). (a) QCNN quantum circuit consisting of d convolutional layers, d pooling layers, and a final fully connected (FC) layer. The measurement of the output qubits labels whether the input state ρ belongs to a given SPT phase. (b) QCNN circuit mimicking renormalization-group flow. In each convolutional layer $f = 1, 2, ..., d$, a disentangling unitary $U_{N/3^{f-1}}^{\dagger}$ and an entangling unitary $U_{N/3^f}$ are applied on sublattices with $N/3^{f-1}$ qubits and $N/3^f$ qubits, respectively. In each pooling layer f, a quantum-error-correction unitary QEC_f is performed on a sublattice with $N/3^{f-1}$ qubits. In the fully connected layer, a disentangling unitary $U_{N/3^d}^{\dagger}$ is applied. At the end, $\lfloor N/3^d \rfloor$ qubits are measured in the X bases. (c) QCNN circuit equivalent to (b) consisting of a constant-depth quantum circuit $U_N^{\dagger}$, the measurement of all qubits in the X basis and classical postprocessing. The label of the quantum phase is determined as a Boolean function $G(x)$ of the measured bit strings x.

where $K_l \in \{\sqrt{p_{\mathbb{1}}}\mathbb{1}, \sqrt{p_X}X, \sqrt{p_Y}Y, \sqrt{p_Z}Z\}^{\otimes N}$ are Kraus operators, p_E are probabilities of Pauli errors $E = X, Y, Z$ and $p_{\mathbb{1}} + p_X + p_Y + p_Z = 1$. For $p_X = p_Y = p_Z$, this error channel describes single-qubit depolarizing noise.

We formulate quantum phase recognition on NISQ devices as a task to identify whether the exact ground state $|\psi\rangle$ of the Hamiltonian (1) belongs to a given quantum phase provided access only to the noisy state ρ, which approximates $|\psi\rangle$.

III. QUANTUM CONVOLUTIONAL NEURAL NETWORKS

Our goal is to design QCNNs that detect the SPT phases of the generalized cluster-Ising model via quantum phase recognition. To perform quantum phase recognition, we process the ground states ρ with the QCNN depicted in Fig. 1(a) consisting of d convolutional layers, d pooling layers, and a final fully connected layer. In each convolutional layer $f = 1, 2, ..., d$, a translationally invariant unitary V_f is applied. In a pooling layer, the system size is reduced by measuring a fraction of qubits and applying feed-forward gates W_f conditioned on the measurement outcomes on the remaining qubits. In this paper, we consider the reduction of system size by a factor of three in each pooling layer. As a result, the maximal depth $d = \lfloor \log_3 N \rfloor$ of the QCNN is logarithmic in system size N. In the fully connected layer, a general unitary V_{FC} is performed on all remaining qubits and the qubits are read out labeling whether the ground state $|\psi\rangle$ belongs to a given SPT phase or not.

For each SPT phase, we construct the QCNN depicted in Fig. 1(b) by generalizing the procedure proposed in Ref. [14]. First, we identify a characteristic state belonging to each SPT phase. For the ZXZ ($ZXXXZ$) SPT phase this is the ZXZ ($ZXXXZ$) cluster state, which can be mapped onto a product

state by a disentangling unitary $U^{\dagger}$ consisting of two (four) layers of two-qubit gates between neighboring qubits, see Appendix B for details. The convolutional layers of the QCNN consist of the disentangling unitary $U_{N/3^{f-1}}^{\dagger}$ mapping the corresponding cluster state on $N/3^{f-1}$ qubits onto a product state and the entangling unitary $U_{N/3^f}$ mapping the product state on a sublattice with $N/3^f$ qubits onto the cluster state. As a result, we obtain the cluster state for a reduced system size after the measurement of the remaining qubits in each pooling layer. By construction, the cluster state is a fixed point of the QCNN circuit.

Next, we make all states belonging to the ZXZ ($ZXXXZ$) SPT phase flow towards the ZXZ ($ZXXXZ$) cluster state with the increasing depth of the QCNN. To this end, we implement in pooling layers a procedure that is analogous to quantum error correction (QEC), identifying perturbations away from the cluster state as errors. These errors are detected by measurements in the pooling layers and corrected by feed-forward gates W_f on the remaining qubits, which are conditioned on the measurement outcomes. A measurement and a feed-forward gate can be replaced by an entangling gate and tracing out of the "measured" qubits. Using this equivalence, we represent the QEC procedure in each pooling layer f as a unitary QEC_f as depicted in Fig. 1(b). It has been shown in Ref. [14] that by correcting X_j and X_jX_{j+1} errors one can make all pure ground states of the cluster-Ising Hamiltonian belonging to the ZXZ SPT phase (for $J_2 = 0$) flow towards the ZXZ cluster state. In this way, the QCNN mimics a renormalization-group flow [22].

In the fully connected layer, we measure stabilizer elements, i.e., either C_j or D_j for the ZXZ phase or the $ZXXXZ$ SPT phase, respectively. This measurement is performed by applying the disentangling unitary $U_{N/3^d}^{\dagger}$ and reading out all

remaining qubits in the X basis. For system size N and depth d, we have $m = \lfloor N/3^d \rfloor$ output qubits. The QCNN output

$$y = \frac{1}{m} \sum_{j=-(m-1)/2}^{(m-1)/2} \langle X_{\frac{N+1}{2}+j\times 3^d} \rangle \qquad (5)$$

is thus the expectation value of X averaged over the m output qubits.

Before discussing the performance of the constructed QC-NNs in the presence of noise due to decoherence and gate infidelities on NISQ devices, we make a crucial observation allowing for a substantial shortening of the QCNN circuits. A large part of the QCNN circuits depicted in Fig. 1(b) can be efficiently implemented in classical postprocessing if the QEC procedures $\widetilde{\text{QEC}}_f = U_{N/3^f}^\dagger \text{QEC}_f U_{N/3^f}$ transformed by the entangling unitaries $U_{N/3^f}$ map X-basis eigenstates $|x\rangle$ onto other X-basis eigenstates $|x'\rangle$,

$$\widetilde{\text{QEC}}_f |x\rangle \propto |x'\rangle . \qquad (6)$$

In this case, the QCNNs are equivalent to a constant depth quantum circuit consisting of the disentangling unitary $U_N^\dagger$, the measurement of all qubits in the X basis and classical postprocessing as depicted in Fig. 1(c). See Appendix B for the derivation of these equivalent QCNN circuits. In these equivalent QCNN circuits, only the first convolutional layer is implemented on a quantum computer. The remaining convolutional layers, all pooling layers and the fully connected layer are implemented after the measurement of all qubits in classical postprocessing as a bit-string-valued Boolean function $G(x) = x'$ of the measured bit strings $x = x_1 x_2 \ldots x_N$, where $x_j = 0, 1$ corresponds to measuring $X_j = +1, -1$. Errors perturbing the cluster states lead to flipped measurement outcomes after the disentangling unitary $U_N^\dagger$. These error syndromes are then corrected in classical postprocessing.

Note that the QCNN proposed in Ref. [14] satisfies the condition (6) and its equivalent QCNN circuit consisting of a constant-depth quantum circuit, measurement and classical postprocessing has been developed and experimentally realized in Ref. [42].

In this paper, we consider the equivalent QCNN circuits depicted in Fig. 1(c). First, we numerically obtain the ground states of the Hamiltonian (1) in the thermodynamic limit using the infinite density matrix renormalization group (iDMRG) algorithm [45], see Appendix A for details. Next, we perform the constant-depth quantum circuit on the infinite MPSs by sequentially applying two-qubit gates between neighboring qubits. Then, we sample M_S outcomes of the measurement of N qubits from the infinite MPSs. Finally, we determine the QCNN output y from the measured bit strings x using the Boolean function $G(x)$ as

$$y = \frac{1}{m}\frac{1}{M_S} \sum_{j=-(m-1)/2}^{(m-1)/2} \sum_x [1 - 2\,G(x)_{\frac{N+1}{2}+j\times 3^d}]. \qquad (7)$$

IV. TOLERANCE TO SYMMETRY-PRESERVING ERRORS

NISQ computers operate in the presence of noise due to decoherence and gate infidelities. To enable the exploitation of QCNNs as a characterization tool for NISQ computers, it
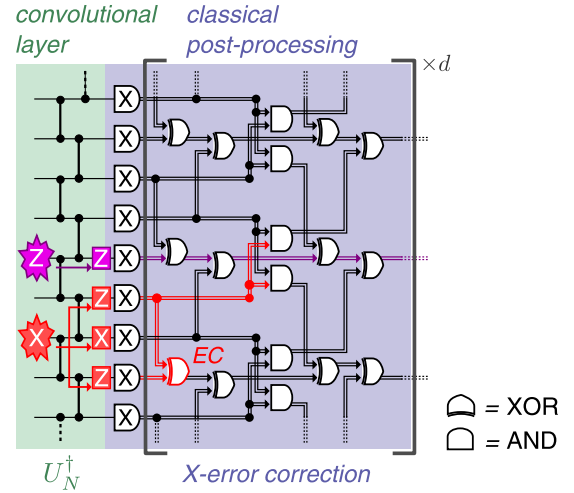

FIG. 2. QCNN with X-error correcting layers detecting the ZXZ phase. The QCNN circuit consists of a constant-depth quantum circuit, the measurement of all qubits in the X basis and classical postprocessing. The quantum circuit performs the disentangling unitary $U_N^\dagger$ consisting of controlled Z gates between neighboring qubits. The outcomes x of the measurement in the X basis are processed by the Boolean function $G(x)$, expressed as a logic circuit in terms of AND and XOR gates. The logic circuit is composed of d layers correcting the syndromes of X errors. Red and purple lines show the propagation of X errors and Z errors, respectively, through the QCNN circuit. The X-error syndrome is corrected by the XOR gate marked in red.

is thus crucial to investigate the effects of noise on the performance of QCNNs and to construct QCNNs whose output is robust against noise.

We expect that the preparation of typical many-body ground states $|\psi\rangle$ will require substantially deeper quantum circuits than the QCNNs considered in this paper, which can be implemented in very short constant depth as discussed above. We thus focus on the robustness of QCNNs against errors that occur during the preparation of many-body ground states $|\psi\rangle$ described by the error channel (4) and neglect errors occurring during the QCNN circuits.

We start by investigating the performance of the QCNN proposed in Ref. [14] in the presence of incoherent X errors described by the error channel (4) with $p_Y = p_Z = 0$. We use the compact implementation as a quantum circuit consisting of the disentangling unitary $U_N^\dagger$, the measurement of all qubits in the X basis and classical postprocessing. We show the QCNN circuit in Fig. 2. The disentangling unitary $U_N^\dagger$ consists of controlled Z gates between neighboring qubits. The outcomes x of the measurement in the X basis are processed by the Boolean function $G(x)$, which is expressed as a logic circuit in terms of AND and XOR gates, see Fig. 2. The key feature of the QCNN is that it identifies and corrects perturbations away from the ZXZ cluster state. In particular, it corrects coherent X_j and $X_j X_{j+1}$ errors, which drive perturbations away from the cluster state to other ground states of the Hamiltonian (1) [14]. The logic circuit is composed of d layers $f = 1, 2, ..., d$, which correspond to the X-error correcting $\widetilde{\text{QEC}}_f$ procedures transformed by the disentangling unitary $U_N^\dagger$.
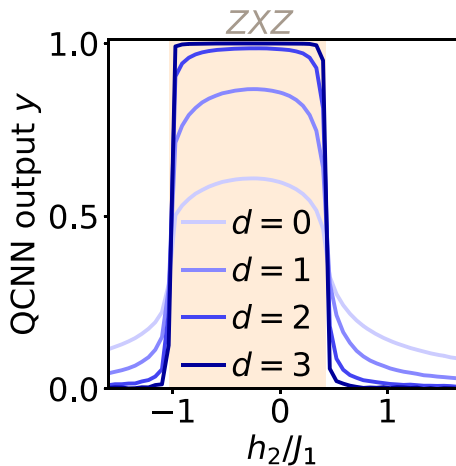
FIG. 3. QCNN consisting of X-error correcting layers for the ground states of the cluster-Ising Hamiltonian (1) perturbed by incoherent X errors. The QCNN output as a function of h_2/J_1 and different depths d of the QCNN. The orange region denotes the ZXZ phase. (Parameters: $h_1/J_1 = 0.5$, $J_2 = 0$, $N = 1215$, $p_X = 0.1$, $p_Y = p_Z = 0$, $M_S = 10^4$).

We plot in Fig. 3 the QCNN output across a cut through the phase diagram as a function of h_2/J_1 for fixed $h_1/J_1 = 0.5$ and $J_2 = 0$ for different depths d of the QCNN. We can see that the QCNN output converges to unity with the increasing depth of the QCNN in the ZXZ SPT phase but vanishes with the increasing depth outside of the SPT phase. This demonstrates our first observation that QCNNs can tolerate incoherent X errors since their QEC procedures can correct not only coherent perturbations, that transform the cluster state to another ground state in the SPT phase, but also incoherent errors.

The QCNN output converges to ideal noise-free values with increasing depth d for any probability $p_X \neq 0.5$ of incoherent X errors, since the error channel is invertible for these cases, see Appendix C for details. For $p_X = 0.5$ the situation is qualitatively different as the error channel (4) is not invertible. Invertible symmetry-preserving error channels for $p_X \neq 0.5$ preserve SPT order [43]. In contrast, the noninvertible error channel for $p_X = 0.5$ annihilates SOPs

$$\mathcal{E}^\dagger(S_{jk}) = (1 - 2p_X)^2 S_{jk} = 0, \qquad (8)$$

for all j and k, where $\mathcal{E}^\dagger(\mathcal{O}) = \sum_{l=1}^{4^N} K_l^\dagger \mathcal{O} K_l$ is the adjoint channel to Eq. (4). As a result, also the QCNN output vanishes for any input ground state and any depth d.

We conclude that QCNNs recognizing the ZXZ SPT phase can tolerate symmetry-preserving X errors, provided that the error channel is invertible.

V. TOLERANCE TO SYMMETRY-BREAKING ERRORS

Since noise in NISQ devices typically does not preserve the symmetries of problem Hamiltonians, it is important to investigate the robustness of QCNNs against symmetry-breaking errors.

While coherent and incoherent X errors are tolerated by the QCNN designed in Ref. [14], the situation is fundamentally different for incoherent Z errors as they break the $\mathbb{Z}_2 \times \mathbb{Z}_2$

symmetry of the Hamiltonian (1). Z errors described by the error channel (4) with $p_X = p_Y = 0$ lead to a decrease of the SOPs

$$\mathcal{E}^\dagger(S_{jk}) = (1 - 2p_Z)^{\frac{L-1}{2}} S_{jk}, \qquad (9)$$

which scales exponentially with their length $L = k - j + 1$. As a result, the SOPs rapidly vanish with the increasing length L for any finite (nonunity) Z-error probability $p_Z \neq 0, 1$.

Similarly to SOPs, the original design [14] of the QCNN depicted in Fig. 2 is substantially affected by Z errors. The syndrome of a Z_j error, i.e., the flipped outcome x_j of the measurement in the X basis, is denoted in Fig. 2 by a purple line. In contrast to X-error syndromes that are corrected (see red lines in Fig. 2), Z-error syndromes (purple line) propagate through the QCNN circuit, see Appendix D for more details. As the system size is reduced by a factor of three in each layer, the density of Z-error syndromes increases with the increasing depth of the QCNN. As a result, the output of the QCNN rapidly decreases with the depth d both in the ZXZ SPT phase and outside of the phase for any finite probability $p_Z \neq 0, 1$, see Appendix E for details.

To perform quantum phase recognition on NISQ devices, QCNNs thus need to be robust against symmetry-breaking errors. In Ref. [42], symmetry-breaking errors were corrected in the fully connected layer. While this improved the robustness of the QCNN with depth $d = 1$, this approach is impractical for deeper QCNNs due to the rapidly increasing density of Z errors with d. To overcome this issue, we construct a new QCNN depicted in Fig. 4 by alternating the original X-error correcting layers with new Z-error correcting layers. The Z-error correcting layer f consists of a new QEC procedure that can be efficiently implemented in classical postprocessing as the majority function

$$M(x_{j-7\times 3^{f-1}}, x_j, x_{j+7\times 3^{f-1}})$$
$$= [(x_{j-7\times 3^{f-1}} \oplus x_j) \wedge (x_j \oplus x_{j+7\times 3^{f-1}})] \oplus x_j, \qquad (10)$$

where $\wedge$ is the AND gate and $\oplus$ is the XOR gate, see Fig. 4. The majority function $M(x_{j-7\times 3^{f-1}}, x_j, x_{j+7\times 3^{f-1}})$ returns the value of the majority of the three bits $x_{j-7\times 3^{f-1}}$, x_j, and $x_{j+7\times 3^{f-1}}$. It thus removes isolated error syndromes, see purple lines in Fig. 4 and Appendix D for more details. The corresponding QEC unitary is described in Appendix B.

We start by investigating the QCNN with alternating X- and Z-error correcting layers for the ZXZ cluster state perturbed by incoherent Z errors as the input state. We plot in Fig. 5 the QCNN output as a function of the Z-error probability p_Z. We can see an alternating QCNN output after odd and even layers. Z errors propagate through odd, X-error correcting layers and, as the system size is reduced by a factor of three, the density of Z errors increases. This error concentration leads to the decrease of the QCNN output after odd layers, compare blue and red lines in Fig. 5. In contrast, even layers correct Z errors leading to the decrease of their density and the increase in the QCNN output. We find that the QCNN can tolerate Z errors for error probabilities p_Z below a threshold $p_{th} = 0.054$ as the error correction in even layers dominates over the error concentration in odd layers leading to a net increase of the QCNN output after every two layers, see blue lines in Fig. 5. On the other hand, Z errors cannot
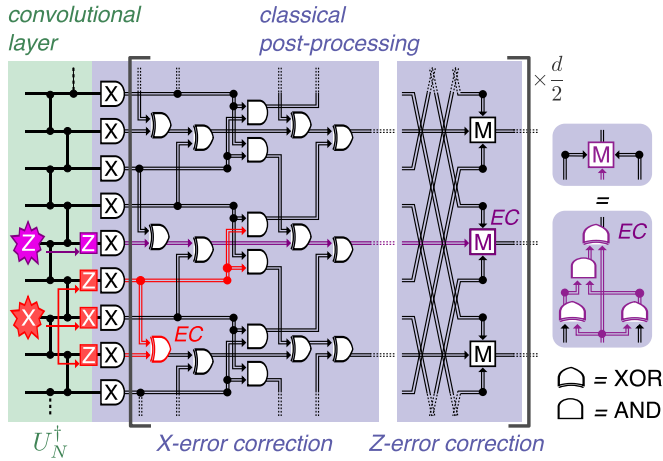
FIG. 4. QCNN with alternating layers correcting X errors and Z errors for detecting the ZXZ phase. The QCNN circuit consists of a constant-depth quantum circuit, the measurement of all qubits in the X basis and classical postprocessing. The quantum circuit performs the disentangling unitary $U_N^\dagger$ consisting of controlled Z gates between neighboring qubits. The outcomes x of the measurement in the X basis are processed by the Boolean function $G(x)$ expressed as a logic circuit in terms of AND and XOR gates as well as of the majority function M. The decomposition of the majority function M into AND and XOR gates is shown on the right. Red and purple lines show the propagation of X errors and Z errors, respectively, through the QCNN circuit. The logic circuit consists of d layers $f = 1, 2, ..., d$. Odd layers f correct the syndromes of X errors (see the XOR gate marked in red) and even layers f correct the syndromes of Z errors (see the majority function M marked in purple).

be tolerated above the threshold since the error concentration dominates over the error correction leading to a net decrease of the QCNN output. See Appendix D for the derivation of the threshold error probability p_{th}. This shows that implementing a Z-error correcting layer after each X-error correcting layer
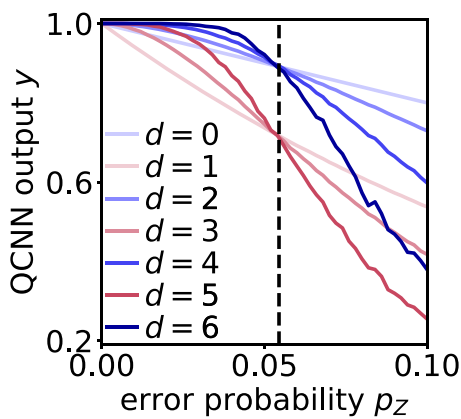

FIG. 5. QCNN consisting of alternating layers correcting X errors and Z errors for the ZXZ cluster state perturbed by symmetry-breaking Z errors. The QCNN output as a function of the error probability p_Z of Z errors for different depths d. The black dashed line shows the threshold error probability $p_{th} = 0.054$. (Parameters: $N = 1215$, $M_S = 10^4$, $p_X = p_Y = 0$).

prevents the concentration of symmetry-breaking Z errors below the threshold error probability.

We now study the error tolerance of QCNNs with alternating layers for different ground states of the cluster-Ising Hamiltonian. We consider a depolarizing channel with $p_X = p_Y = p_Z$ describing the presence of X errors, Z errors, and their simultaneous appearance $Y = iXZ$, representing a typical situation for NISQ devices. We plot in Fig. 6(a) the QCNN output for different ground states as a function of h_2/J_1 and different depths d of the QCNN, obtained using infinite MPS simulations. We can see that the QCNN tolerates the incoherent errors as the QCNN output converges to unity with the increasing depth d in the SPT phase and it vanishes outside of the SPT phase [46].

The two types of layers in the QCNN play complementary roles. The X-error correcting layers implement renormalization-group flow with states belonging to the ZXZ phase flowing towards the ZXZ cluster state and states outside of the ZXZ phase diverging from it. The X-error correcting layers are thus crucial for recognizing the ZXZ phase. In contrast, the Z-error correcting layers reduce the density of error syndromes by removing syndromes due to symmetry-breaking errors. As a result, the Z-error correcting layers equip the QCNNs with the tolerance to symmetry-breaking errors, see Fig. 6(a).

We now show that the QCNNs with alternating X- and Z-error correcting layers can perform phase recognition provided that the probability of errors in the prepared states is below the threshold p_{th}. To this end, we plot the QCNN output for the depth $d = 4$ as a function of h_1/J_1 and h_2/J_1 in Fig. 6(b) in the presence of depolarizing noise. We can see that the QCNN output attains near unity value in the ZXZ phase and vanishing value outside of the ZXZ phase. The abrupt change of the QCNN output from near unity values to vanishing values coincides with the phase boundary (black crosses) determined by iDMRG simulations, see Appendix A for more details.

In contrast to the QCNN, SOPs S_{jk} are significantly suppressed in the presence of incoherent errors. We plot in Fig. 6(c) SOPs S_{jk} as a function of h_2/J_1 for different lengths L in the presence of depolarizing noise. We can see that the SOPs rapidly vanish both in the ZXZ SPT phase and outside of the phase with the increasing length L.

In conclusion, the QCNN constructed here recognizes the ZXZ SPT phase in the presence of symmetry-breaking errors below the threshold error probability p_{th}. In contrast, SOPs rapidly vanish with the increasing length for any finite probability of symmetry-breaking errors. The previously considered QCNN of Ref. [14] cannot tolerate symmetry-breaking errors either as its output decreases with the increasing depth for any error probability. As a result, it cannot recognize the SPT phase in the presence of symmetry-breaking errors.

VI. $ZXXXZ$ SYMMETRY-PROTECTED TOPOLOGICAL PHASE

We now discuss the extension of phase recognition capabilities of the error-tolerant QCNNs we introduced to distinguish the $ZXXXZ$ SPT phase from topologically trivial phases as well as the ZXZ and $ZXXXZ$ SPT phases from one another.
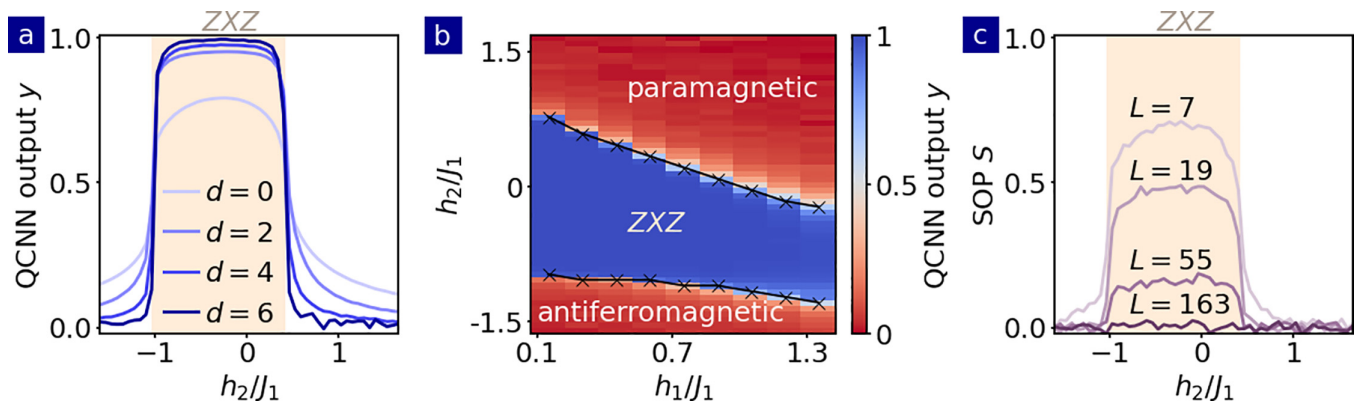
FIG. 6. Recognition of ZXZ SPT phase by QCNN consisting of alternating layers correcting X errors and Z errors for ground states of the Hamiltonian (1) perturbed by depolarizing noise. (a) The QCNN output y on the cut through the phase diagram as a function of h_2/J_1 for different depths d of the QCNN. (b) The QCNN output y as a function of h_1/J_1 and h_2/J_1 for the depth $d = 4$. Black crosses show the phase boundary identified using iDMRG simulations. (c) String order parameters (SOPs) S_{jk} on the cut through the phase diagram as a function of h_2/J_1 for different lengths $L = k - j + 1$. The orange regions denote the ZXZ phase. [Parameters: $p_X = p_Y = p_Z = 0.015$, $M_S = 10^4$, $J_2 = 0$; (a) $N = 1215$, $h_1/J_1 = 0.5$; (b) $N = 135$; (c) $h_1/J_1 = 0.5$].

Similarly, as for the ZXZ SPT phase, we construct a QCNN that detects the $ZXXXZ$ phase from the topologically trivial paramagnetic and antiferromagnetic phases. Now the convolutional layer consists of a disentangling unitary $\tilde{U}_N^\dagger$ mapping the $ZXXXZ$ cluster state onto a product state. The QEC procedures are amended to correct X_j and $X_j X_{j+1}$ errors perturbing the $ZXXXZ$ cluster state, see Appendix F for more details about the QCNN for the $ZXXXZ$ phase. To equip the QCNN with the tolerance to state preparation errors, we follow the same approach as in the previous section. We alternate X-error correcting layers and Z-error correcting layers. While the X-error correcting layers had to be amended for the target $ZXXXZ$ phase, we can employ the same procedure based on the majority function of Eq. (10) to correct Z_j errors. In contrast to the disentangling circuit $U_N^\dagger$ for the ZXZ cluster state, which commutes with Z_j errors, the disentangling unitary $\tilde{U}_N^\dagger$ for the $ZXXXZ$ cluster state maps Z_j errors onto the errors $\tilde{U}_N^\dagger Z_j \tilde{U}_N = Y_{j-1} Z_j Y_{j+1}$, which flip the measurement outcomes x_{j-1}, x_j and x_{j+1} on the three qubits $j - 1$, j and $j + 1$. Due to this multiplication of symmetry-breaking errors, the threshold probability $\tilde{p}_{th} = 0.018$ is reduced compared to $p_{th} = 0.054$ for the ZXZ cluster state.

We determine the phase boundary between the ZXZ phase and the $ZXXXZ$ phase to be located at $J_1/J_2 = 0.95$ via iDMRG simulations.

We start with a QCNN recognizing the ZXZ phase from the $ZXXXZ$ phase. Before showing the results, we explain the construction of this QCNN, which requires identifying the perturbations driving the ground states for nonvanishing h_2 and J_2 away from the characteristic ZXZ cluster state. These perturbations include the $X_j X_{j+1}$ interactions and the stabilizer elements D_j. The $X_j X_{j+1}$ interactions are corrected by the original X-error correcting procedure. The D_j stabilizer elements are mapped by the disentangling unitary onto $U_N^\dagger D_j U_N = -Y_{j-1} X_j Y_{j+1}$, which lead to the same syndromes after the measurement of all qubits in the X basis (flipped measurement outcomes at qubits $j - 1$ and $j + 1$) as X_j perturbations, for which $U_N^\dagger X_j U_N = Z_{j-1} X_j Z_{j+1}$. As a result, the

QCNN depicted in Fig. 2 constructed in the previous section for correcting X_j and $X_j X_{j+1}$ perturbations, corrects D_j perturbations as well and can be readily used to recognize the ZXZ phase from the $ZXXXZ$ phase. To achieve tolerance to state preparation errors on NISQ devices, we can thus alternate the X-error correcting layers with Z-error correcting layers in the same way as depicted in Fig. 4.

We plot the QCNN output (blue lines) as a function of J_1/J_2 in Fig. 7 for different depths of the QCNN in the presence of depolarizing noise. We can see that the QCNN detects the ZXZ phase as its output converges to unity in the phase ($J_1/J_2 > 0.95$) and vanishes in the $ZXXXZ$ phase ($J_1/J_2 < 0.95$).
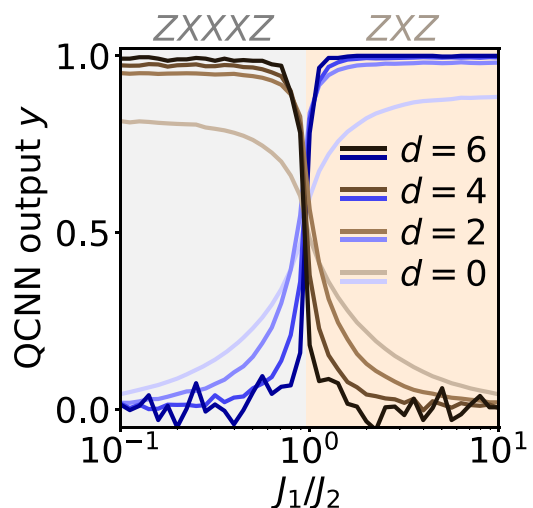
FIG. 7. Distinguishing the ZXZ SPT phase and the $ZXXXZ$ SPT phase. The QCNN output as a function of J_1/J_2 for different depths d of the QCNN for ground states perturbed by depolarizing noise. The QCNN with alternating X-error correcting layers and Z-error correcting layers (blue lines) as well as the QCNN with alternating C-error and Z-error correcting layers (brown lines). (Parameters: $N = 1215$, $M_S = 10^3$, $h_1/J_2 = 0$, $h_2/J_2 = 0.1$, $p_X = p_Y = p_Z = 0.01$).
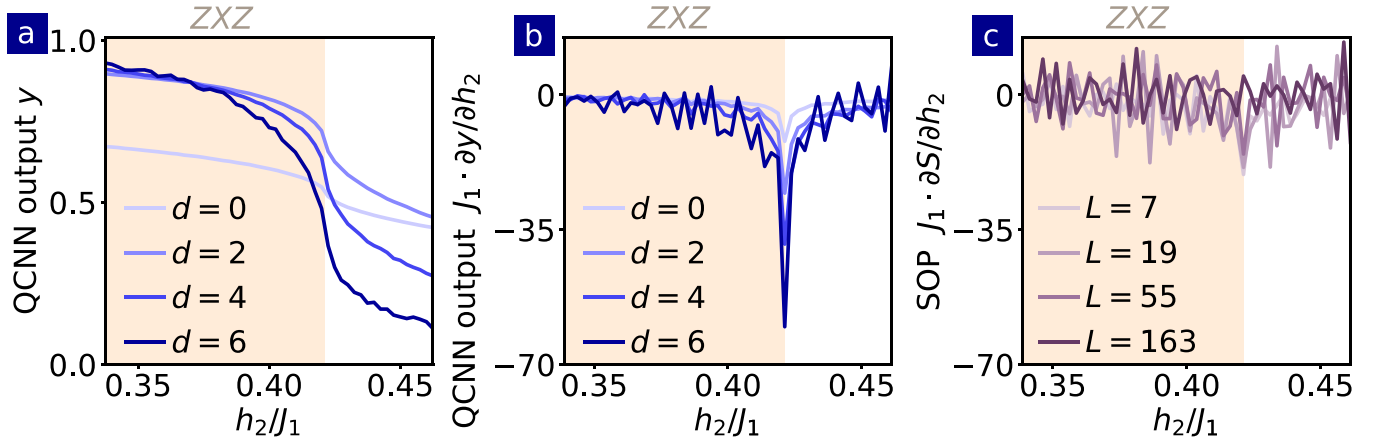
FIG. 8. Detecting the phase boundary between the ZXZ phase and the paramagnetic phase in the presence of depolarizing noise. (a) The output of the QCNN with alternating layers correcting X errors and Z errors close to the phase boundary as a function of h_2/J_1 for different depths d. (b) The slope of the QCNN output $\partial y/\partial h_2$ with respect to the Hamiltonian parameter h_2. (c) The slope of SOPs $\partial S_{jk}/\partial h_2$ with respect to the Hamiltonian parameter h_2 close to the phase boundary as a function of h_2/J_1 for different lengths $L = k - j + 1$. The orange regions denote the ZXZ phase. [Parameters: $M_S = 10^4$, $h_1/J_1 = 0.5$, $J_2 = 0$, $p_X = p_Y = p_Z = 0.015$; (a) and (b) $N = 1215$].

We now discuss the construction of a QCNN recognizing the $ZXXXZ$ phase from the ZXZ phase. Here, the stabilizer elements C_j and $X_j X_{j+1}$ interactions play the role of perturbations away from the $ZXXXZ$ cluster state. The disentangling unitary $\tilde{U}_N^\dagger$ for the $ZXXXZ$ cluster state maps the C_j perturbations onto $\tilde{U}_N^\dagger C_j \tilde{U}_N \propto Y_{j-1} X_j Y_{j+1}$. The C_j perturbations have different syndromes after the measurement of all qubits in the X basis than X_j and $X_j X_{j+1}$ perturbations, $\tilde{U}_N^\dagger X_j \tilde{U}_N = Y_{j-2} X_{j-1} X_j X_{j+1} Y_{j+2}$ and $\tilde{U}_N^\dagger X_j X_{j+1} \tilde{U}_N = Y_{j-2} Z_{j-1} Z_{j+2} Y_{j+3}$. As a result, we need to amend the QEC procedures to correct the C_j perturbations, see Appendix B for more details about this procedure. The C-error correcting procedure also corrects $X_j X_{j+1}$ perturbations and X_j perturbations, where the latter now come about only due to noise on NISQ devices, see Appendix F for more details. To achieve tolerance to incoherent Z_j errors, we alternate the C-error correcting layers with the Z-error correcting layers based on the majority function. We plot the resulting QCNN output (brown lines) as a function of J_1/J_2 in Fig. 7 for different depths of the QCNN in the presence of depolarizing noise. We can see that the QCNN detects the $ZXXXZ$ phase as its output converges to unity in the phase $J_1/J_2 < 0.95$ and vanishes in the ZXZ phase $J_1/J_2 > 0.95$.

We have thus demonstrated that the error-tolerant QCNNs we introduced can distinguish not only topological phases from topologically trivial phases but also two topological phases from one another. To this end, the QCNN for the $ZXXXZ$ phase needed to be amended to correct C_j perturbations whereas the original QCNN for the ZXZ phase was already capable of correcting D_j perturbations. While the QCNNs for the ZXZ phase and the $ZXXXZ$ phase are designed to correct different symmetry-preserving errors, the tolerance to symmetry-breaking errors is achieved in a universal manner in all QCNNs by employing the same Z-error correcting layer based on the majority function.

VII. PHASE BOUNDARY

So far, we have shown that the QCNNs we consider can recognize SPT phases in the presence of incoherent errors. We now investigate the tolerance of incoherent errors close to phase boundaries. Precisely detecting phase boundaries is one of the major challenges of many-body physics due to diverging correlation lengths and the rapid growth of entanglement in their vicinity [22,47].

In Fig. 8(a) we plot the output of the QCNN for the ZXZ phase close to a phase boundary between the ZXZ phase and the paramagnetic phase in the presence of depolarizing noise. We can see that the QCNN tolerates incoherent errors well in the SPT phase as its output converges to unity with the increasing depth d. On the other hand, close to the phase boundary the QCNN does not tolerate incoherent errors as its output decreases with the increasing depth d. We thus observe that while symmetry-preserving X errors can be tolerated for any ground state, the tolerance to symmetry-breaking errors is limited close to phase boundaries.

We now further quantify the behavior close to the phase boundary between the ZXZ phase and the paramagnetic phase. We investigate, which probabilities p_Z of symmetry-breaking Z errors can be tolerated for each ground state belonging to the ZXZ phase. To do so, we determine the threshold error probability p_{th} for various ground states below which the QCNN output converges to unity with an increasing depth d. Above p_{th} the output decreases with d. We plot in Fig. 9 the threshold error probability p_{th} (red dots) as a function of the correlation length ξ of the ground states [48]. We can see that the threshold error probability decreases with the correlation length. We fit this decrease by the exponential function $p_{th} = p_{th}^0 \exp(-\xi/\bar{\xi})$ with the fitted parameters p_{th}^0 and $\bar{\xi}$ in Table I. We can see in Fig. 9 a similar exponential decrease of the threshold probability p_{th} close to the phase boundaries between the $ZXXXZ$ phase and the paramagnetic state (gray diamonds) as well as between the $ZXXXZ$ phase
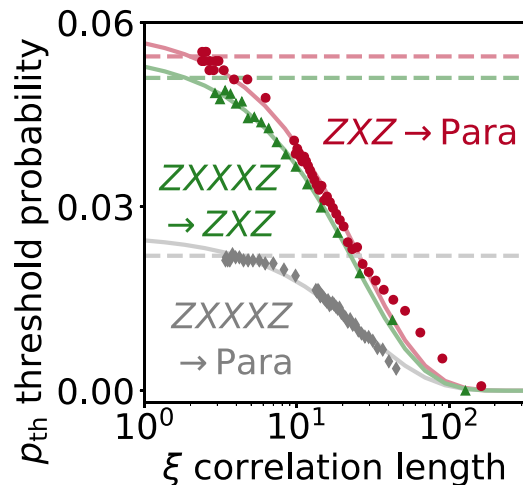
FIG. 9. Tolerance to incoherent errors close to phase boundaries. The plot shows the threshold probability p_{th} of symmetry-breaking errors as a function of the correlation length ξ in the vicinity of phase boundaries between the ZXZ phase and the paramagnetic phase (red dots), the $ZXXXZ$ phase and the paramagnetic phase (gray diamonds), and the $ZXXXZ$ phase and the ZXZ phase (green triangles). Solid lines show fitted exponential functions and dashed lines show threshold error probabilities of corresponding cluster states. [Parameters: $N = 1215$, $M_S = 10^5$, $h_1/J_1 = 0.5$ and $J_2 = 0$ (red dots), $h_1/J_2 = 0.5$ and $J_1 = 0$ (gray diamonds), $h_1 = 0$ and $h_2/J_2 = 0.1$ (green triangles)].

and the ZXZ phase (green triangles). The error tolerance is thus strongly suppressed close to all phase boundaries when the correlation length exceeds the characteristic value $\bar{\xi} \approx 26$, cf. Table I.

To identify the phase boundary between the ZXZ phase and the paramagnetic phase in the presence of symmetry-breaking errors, we plot in Fig. 8(b) the slope $\partial y/\partial h_2$ of the QCNN output y with respect to the Hamiltonian parameter h_2 for different depths d of the QCNN. We can see a sharp dip in the slope of the QCNN output precisely located at the phase boundary. The dip becomes more pronounced with the increasing depth d of the QCNN. This shows that while the QCNN output decreases with the increasing depth d close to the phase boundary, see Fig. 8(a), we can still precisely identify the phase boundary as a sharp dip in the slope of the QCNN output. We can see in Fig. 6(b) that the abrupt change of the QCNN output coincides with the phase boundary (black crosses) determined using iDMRG in the entire phase diagram ($J_2 = 0$). We also observe that the slope of the QCNN output exhibits a sharp dip (or peak) at all other phase boundaries of

TABLE I. Fitted parameters of the threshold probability $p_{\text{th}} = p_{\text{th}}^0 \exp(-\xi/\bar{\xi})$ of symmetry-breaking errors as a function of the correlation length ξ close to different phase boundaries.

Phase boundary	p_{th}^0	$\bar{\xi}$
$ZXZ \rightarrow$ Paramagnetic	0.0588 ± 0.0005	26.1 ± 0.6
$ZXXXZ \rightarrow$ Paramagnetic	0.0254 ± 0.0002	28.1 ± 0.6
$ZXXXZ \rightarrow ZXZ$	0.0550 ± 0.0004	24.6 ± 0.6

the generalized cluster-Ising model also for $J_2 \neq 0$ (not shown here).

In stark contrast to these characteristics, individual SOPs and their slopes are largely suppressed for any finite probability of symmetry-breaking errors. We plot in Fig. 8(c) the slope $\partial S_{jk}/\partial h_2$ of the SOPs with respect to the Hamiltonian parameter h_2 for different lengths $L = k - j + 1$. We can see that the slope of the SOPs cannot be distinguished from sampling noise for the number of samples $M_S = 10^4$. Crucially, the slope of the SOPs does not become more pronounced with increasing length L. As a result, one cannot use the SOPs to determine the phase boundary in the presence of symmetry-breaking noise.

The sharp dip in the slope of the QCNN output at critical values of Hamiltonian parameters is a unique feature of phase boundaries and it does not appear at the threshold values of probabilities for incoherent errors, see also Appendix D. This feature distinguishes phase boundaries from threshold probabilities of incoherent errors. It thus provides further evidence that the QCNNs we construct perform phase recognition for the error-free ground states while processing only noisy states, which approximate the former.

In conclusion, the tolerance of the considered QCNNs to symmetry-breaking errors is limited close to phase boundaries due to diverging correlation lengths. Nonetheless, we can precisely determine critical values of Hamiltonian parameters as a dip in the slope of the QCNN output. This is in stark contrast to SOPs and their slopes, which rapidly vanish for any finite probability of symmetry-breaking errors and thus cannot be used to identify phase boundaries in the presence of symmetry-breaking errors.

VIII. SAMPLE COMPLEXITY

We now compare QCNNs to the direct measurement of the input state. We focus on sample complexity, which quantifies the number of projective measurements required to identify to which quantum phase the input state belongs. In the absence of noise, QCNNs substantially reduce sample complexity compared to the direct measurement of SOPs [14]. In the presence of symmetry-breaking noise, SOPs vanish and thus they cannot detect the SPT phase. Instead, we focus on the observable measured by QCNNs. This observable is different from any single SOP and it is robust against symmetry-breaking noise. In this section, we discuss this observable for the QCNN that detects the ZXZ phase and compare the cost of directly sampling it from the input state to sampling the QCNN output.

To determine the observable measured by the QCNN with alternating X- and Z-error correcting layers, we represent the QCNN circuit as a unitary U_{QCNN}, see Appendix G for details. The measurement of the Pauli $X_{\frac{N+1}{2}}$ operator at the end of the QCNN circuit corresponds to the measurement of a multiscale SOP,

$$S_{\text{M}} = U_{\text{QCNN}}^{\dagger} X_{\frac{N+1}{2}} U_{\text{QCNN}}$$
$$= \sum_{ij} \eta_{ij}^{(1)} S_{ij} + \sum_{ijkl} \eta_{ijkl}^{(2)} S_{ij} S_{kl} + ..., \qquad (11)$$

on the input state. The multiscale SOP S_M is a sum of products of SOPs S_{ij} at different lengths $L = j - i + 1$. The length of the SOPs, $L \sim 3^d$, increases exponentially with the depth d of the QCNN. Compared to the QCNN of Ref. [14], the change in the coefficients $\eta^{(n)}$ due to our construction equips the QCNN with error tolerance. The multiscale SOP in Eq. (11) involves at least $2^{3^{d-2}}$ products of SOPs.

As an alternative to executing the QCNN, we can determine the expectation value of the multiscale SOP using the direct measurements of the input state without performing any quantum circuit. Assuming that only measurements in local Pauli bases can be directly performed, which is the case for most devices, we show in Appendix H that the multiscale SOP involves at least $3^{3^{d-4}}$ products of SOPs, which cannot be simultaneously measured via local Pauli measurements as they require sampling in mutually incompatible Pauli bases. As a result, the sample complexity of the direct Pauli measurement scales double exponentially with the depth of the QCNN, corresponding to an exponential scaling with system size for the maximal depth $d = \lfloor \log_3 N \rfloor$. Instead of the direct Pauli measurement, one could estimate the products of SOPs via classical shadow tomography [5]. However, the sample complexity of classical shadow tomography also scales exponentially with the length L of the SOPs. In contrast, all products of SOPs are sampled simultaneously in the QCNN after performing the disentangling unitary. The QCNN thus determines the expectation value of the multiscale SOP with a constant sample complexity in system size (and depth of the QCNN), which exponentially reduces the sample complexity compared to direct Pauli measurements.

Importantly, the equivalent QCNN circuit depicted in Fig. 4, which is based on a constant-depth quantum circuit, measurement and classical postprocessing, measures the multiscale SOP with the same sample complexity as the full quantum QCNN circuit. The constant-depth quantum circuit allows us to simultaneously measure all stabilizer elements C_j. From measured bit strings, we then determine the expectation value of the multiscale SOP in classical postprocessing with the same sample complexity as for the full quantum QCNN circuit.

QCNNs detecting the *ZXXXZ* phase also measure multiscale SOPs, which are sums of double exponentially many products of SOPs T_{jk}. Similarly to the QCNN for the *ZXZ* phase, these QCNNs also reduce sample complexity exponentially compared to direct local Pauli measurements.

IX. CONCLUSIONS

We constructed QCNNs that tolerate incoherent errors due to decoherence and gate infidelities during the preparation of their input states. These QCNNs tolerate symmetry-breaking errors below a threshold error probability in contrast to previous QCNN designs and SOPs, which are significantly suppressed for any nonvanishing error probability. Moreover, their output is robust against invertible symmetry-preserving error channels. The error tolerance is limited close to phase boundaries as the threshold error probability decreases with diverging correlation lengths. However, a steep gradient of the QCNN output at phase boundaries between SPT phases and topologically trivial phases as well as between two SPT

phases allows us to precisely determine critical values of the Hamiltonian parameters.

The QCNN quantum circuits constructed here can be shortened from logarithmic depth in input size to short, constant depth by performing a large part of computation in classical postprocessing after the measurement of all qubits. This substantially improves the performance of QCNNs under NISQ conditions by reducing the number of finite-fidelity quantum gates. The classical postprocessing part of QCNNs consists of logic circuits with at most logarithmic depth in input size. The QCNNs we constructed reduce sample complexity exponentially in input size in comparison to the direct sampling of the QCNN output using local Pauli measurements.

Our paper provides insights into SPT order in open quantum systems, which are subject to decoherence and dissipation. Apart from NISQ computers, the error channel we consider, see Eq. (4), describes typical open quantum systems [43]. On the one hand, SOPs rapidly vanish with an increasing length for any symmetry-breaking error channel as shown in Ref. [43]. On the other hand, our results show that SPT order is not completely washed out for probabilities of symmetry-breaking errors below a finite threshold. This distinction emerges because the multiscale SOPs, that are efficiently measured by the QCNNs we introduce, exploit information about SPT order at different length scales to detect SPT phases in the presence of symmetry-breaking noise.

Due to the tolerance of errors and the short depth of their quantum circuits, the QCNNs constructed here can be readily realized on current NISQ computers to efficiently measure characteristic nonlocal observables of SPT phases. This will facilitate the investigation of topological quantum phases of matter on quantum computers.

As a next step, the construction of QCNN circuits mimicking renormalization-group flow depicted in Fig. 1(b) could be extended for two- and higher-dimensional systems, which can, in addition to SPT phases, feature intrinsic topological order [49] and symmetry enriched topological phases [50]. Using preparation circuits for characteristic ground states belonging to SPT phases [51] as well as symmetry enriched topological phases [52], these ground states can be considered as fixed points of QCNNs and the QEC unitaries in pooling layers can be constructed in analogy to renormalization-group decoders [53].

QCNN circuits for two- and higher-dimensional systems can be equipped with the tolerance to incoherent errors by using our approach. To this end, the QCNN layers mimicking renormalization-group flow are alternated with layers correcting symmetry-breaking errors. While the layers mimicking renormalization-group flow need to be specifically designed for each target quantum phase and given Hamiltonian perturbations, the correction of symmetry-breaking errors can be implemented in a universal manner using the majority function. In this way, all errors perturbing the characteristic state are corrected provided that the error density is small enough, as we demonstrated for different SPT phases of the generalized cluster-Ising model. The majority function provides a universal tool to correct symmetry-breaking errors in QCNNs for two- and higher-dimensional systems.

Other interesting future directions include detecting less understood topological phases such as anyonic chains [54]

and quantum spin liquids [55]. Another promising direction is the training of QCNNs based on parametrized quantum circuits to identify nonlocal observables characterizing topological phases from training data [14,39,40,56].

The QCNNs we constructed open the way for efficiently characterizing noisy quantum data produced by near-term quantum hardware. In addition to the recognition of topological phases, reducing the sample complexity of nonlocal observables will substantially speed up other quantum algorithms. A prominent example is the variational quantum eigensolver for quantum chemistry problems, which involves many repetitions of demanding measurements of molecular Hamiltonians [57,58].

APPENDIX A: NUMERICAL SIMULATIONS

Our main results are based on MPS simulations implemented using the Python library TeNPy [45]. Using iDMRG with the maximal bond dimension $\chi = 150$, we obtain numerically exact ground states $|\psi\rangle$ of the Hamiltonian (1) in the thermodynamic limit to avoid finite size effects. First, we identify phase boundaries as sharp peaks in the second derivative of the ground-state energy with respect to h_2/J_1 for constant h_1/J_1 and $J_2 = 0$ in Figs. 3, 6, 8, with respect to J_1/J_2 for constant $h_1 = 0$ and $h_2/J_2 = 0.1$ in Fig. 7, as well as with respect to h_2/J_2 for constant h_1/J_2 and $J_1 = 0$ in Fig. 14 below.

We implement the QCNN circuits depicted in Fig. 1(c) consisting of a constant-depth quantum circuit, the measurement of all qubits in the X basis and classical postprocessing. The constant-depth quantum circuit performs the disentangling unitary $U_N^\dagger$ consisting of nearest-neighbor two-qubit gates, which can be efficiently applied on the MPSs $|\psi\rangle$ obtained using iDMRG. We simulate the measurement outcomes of N qubits by sampling spin configurations x in the X basis from their probability distribution $P_x = \text{Tr}[|x\rangle\langle x|U_N^\dagger|\psi\rangle\langle\psi|U_N]$ corresponding to the MPS $U_N^\dagger|\psi\rangle$ after having performed the disentangling unitary $U_N^\dagger$. QCNN outputs are determined from the sampled bit strings x as a Boolean function, which is expressed as a logic circuit, see Figs. 2 and 4.

To explore incoherent errors using the error channel (4), we implement the error channel $\mathcal{E}$ by sampling error events $E_l = K_l/\sqrt{p_l}$ from their probability distribution $p_l = \text{Tr}[K_l^\dagger K_l]/2^N$. The error events E_l are products of Pauli operators, which can be efficiently implemented on the MPSs $|\psi\rangle$. We then sample bit strings x from the joint probability distribution

$$P_x = p_l \text{Tr}[|x\rangle\langle x| U_N^\dagger E_l |\psi\rangle\langle\psi| E_l^\dagger U_N], \quad (A1)$$

which correspond to the measurement outcomes for the noisy state $\mathcal{E}(|\psi\rangle\langle\psi|)$ after having performed the disentangling unitary $U_N^\dagger$.

Increasing the bond dimension to $\chi = 200$ does not lead to a visible change in our findings showing that the MPSs accurately describe the ground states of the Hamiltonian (1) and their processing with the QCNNs.

APPENDIX B: QCNN CIRCUITS

In this Appendix, we describe in detail the QCNN circuits used in this work. We first discuss the QCNN detecting the ZXZ phase. Then we describe the QCNN detecting the $ZXXXZ$ phase. Finally, we discuss the QCNNs distinguishing the two SPT phases from one another.

All QCNNs considered in this paper consist of d convolutional layers, d pooling layers and a fully connected layer as depicted in Fig. 1(a). Each convolutional layer $f = 1, 2, ..., d$ consists of a disentangling unitary $U_{N/3^{f-1}}^\dagger$ on $N/3^{f-1}$ qubits followed by an entangling unitary $U_{N/3^f}$ on a sublattice with $N/3^f$ qubits as depicted in Fig. 1(b). Each pooling layer involves a QEC procedure QEC_f. In the fully connected layer, the disentangling unitary $U_{N/3^d}^\dagger$ is applied and all $\lfloor N/3^d \rfloor$ remaining qubits are measured in the X basis. Note that each QEC_f procedure is preceded by the entangling unitary $U_{N/3^f}$, which is implemented in the preceding convolutional layer, and followed by the disentangling unitary $U_{N/3^f}^\dagger$, which is implemented in the following convolutional layer for $f < d$ and in the fully connected layer for $f = d$, see Fig. 1(b). The QCNN circuit is thus equivalent to a single convolutional layer followed by d pooling layers as depicted in Fig. 10(a). The convolutional layer performs the disentangling unitary $U_N^\dagger$. The pooling layer f involves the QEC unitary $\widetilde{\text{QEC}}_f = U_{N/3^f}^\dagger \text{QEC}_f U_{N/3^f}$ transformed by the entangling unitary $U_{N/3^f}$. Note that in this equivalent quantum circuit, convolutional layers for $f > 1$ are absorbed into the $\widetilde{\text{QEC}}_f$ unitaries in the pooling layers. The disentangling unitary $U_{N/3^d}^\dagger$ from the fully connected layer is also absorbed into the $\widetilde{\text{QEC}}_d$ unitary. The fully connected layer in this equivalent quantum circuit thus consists only of the measurement of remaining qubits in the X basis.

For conciseness, we focus here on the equivalent quantum circuits depicted in Fig. 10(a) as the $\widetilde{\text{QEC}}_f$ procedures transformed by the entangling unitary consist of fewer gates than the bare QEC_f procedures.

We first discuss the QCNN detecting the ZXZ phase. We start with the QCNN consisting of X-error correcting layers depicted in Fig. 10(b). The disentangling unitary $U_N^\dagger$ consists of controlled Z gates between neighboring qubits. The transformed QEC procedure $\widetilde{\text{QEC}}_f^X$ consists of controlled-controlled Z gates $\text{C}_x\text{C}_x\text{Z}$, controlled Z gates C_xZ, and controlled-controlled NOT gates $\text{C}_x\text{C}_x\text{NOT}$ with all controls in the X basis. The error-tolerant design of the QCNN depicted in Fig. 10(c) consists of alternating layers correcting X errors and Z errors. The new Z-error correcting procedure $\widetilde{\text{QEC}}_f^Z$ involves SWAP, C_xZ, and $\text{C}_x\text{C}_x\text{Z}$ gates.

Since all gates are controlled in the X basis and they implement either Pauli X or Pauli Z operations on the target
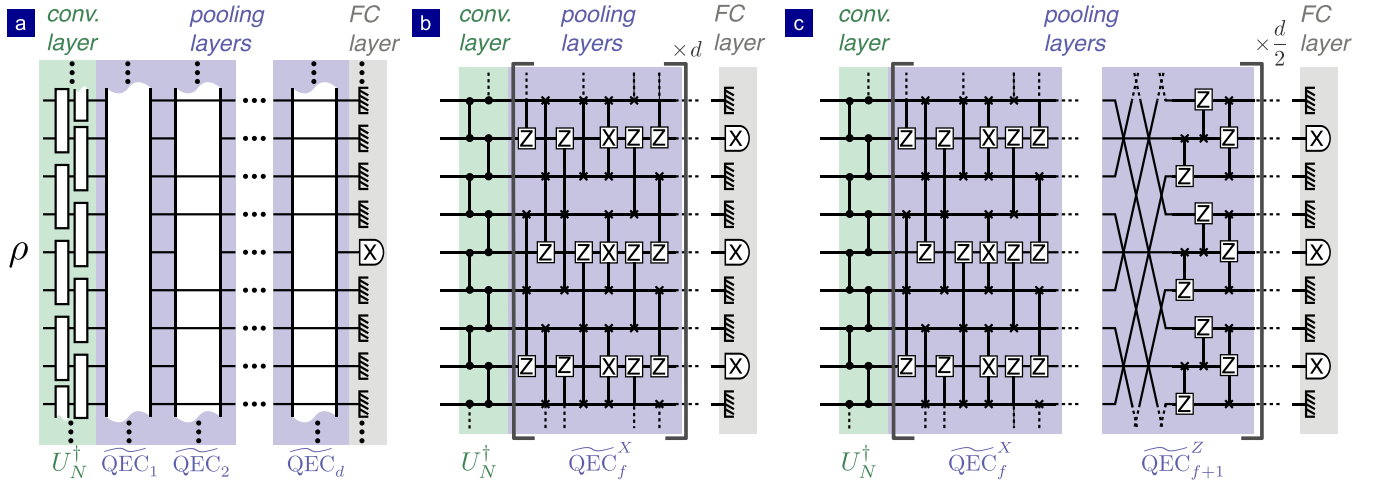
FIG. 10. QCNN quantum circuits. (a) QCNN circuit equivalent to Fig. 1(b) consisting of a single convolutional layer, d pooling layers, and a fully connected (FC) layer. The convolutional layer performs the disentangling unitary $U_N^\dagger$. Each pooling layer $f = 1, 2, ..., d$ performs the $\widetilde{\text{QEC}}_f$ procedure transformed by the entangling unitary $U_{N/3^f}$ on a sublattice with $N/3^{f-1}$ qubits. The fully connected layer involves the measurement of $\lfloor N/3^d \rfloor$ qubits in the X basis. (b) QCNN detecting the ZXZ phase with X-error correcting procedures $\widetilde{\text{QEC}}_f^X$. (c) QCNN detecting the ZXZ phase with alternating procedures $\widetilde{\text{QEC}}_f^X$ and $\widetilde{\text{QEC}}_{f+1}^Z$ correcting X errors and Z errors, respectively. The disentangling unitary $U_N^\dagger$ involves controlled Z gates with controls in the computational basis. The $\widetilde{\text{QEC}}_f^X$ procedure consists of controlled-controlled Z gates $C_x C_x Z$, controlled Z gates $C_x Z$, and controlled-controlled NOT gates $C_x C_x \text{NOT}$ with all controls in the X basis. The $\widetilde{\text{QEC}}_{f+1}^Z$ procedure consists of SWAP, $C_x C_x Z$, and $C_x Z$ gates.

qubit, the $\widetilde{\text{QEC}}_f^X$ and $\widetilde{\text{QEC}}_f^Z$ procedures map X-basis eigenstates $|x\rangle$ onto other X-basis eigenstates $\pm |g_f(x)\rangle$, where $g_f : \{0, 1\}^N \to \{0, 1\}^N$ is a Boolean function. As we also measure in the X basis in the fully connected layer, the processing of a quantum state ρ by the QCNN can be implemented in classical postprocessing as a Boolean function $G(x) = (g_d \circ g_{d-1} \circ ... \circ g_1)(x)$ after measuring all qubits in the X basis. In particular, the output

$$\langle X_j \rangle = \text{Tr}[X_j Q_d U_N^\dagger \rho U_N Q_d^\dagger] = \sum_x P_x (1 - 2G(x)_j), \quad \text{(B1)}$$

of qubit j measured in the fully-connected layer of the full quantum QCNN circuit [Fig. 10(a)] can be determined from bit strings x measured after the convolutional layer by using the jth element of the output of the Boolean function $G(x)$, where $P_x = \text{Tr}[|x\rangle\langle x| U_N^\dagger \rho U_N]$ is the probability of measuring a bit string x after the first convolutional layer and

$$Q_d = \widetilde{\text{QEC}}_d ... \widetilde{\text{QEC}}_2 \widetilde{\text{QEC}}_1. \quad \text{(B2)}$$

We thus only need to apply the single disentangling unitary $U_N^\dagger$ on a quantum computer, measure all qubits in the X basis and determine the QCNN output in classical postprocessing from the measured bit strings x, see Fig. 1(c). The QCNN quantum circuits depicted in Figs. 10(b) and 10(c) can thus be implemented as equivalent circuits depicted in Figs. 2 and 4, respectively. The Boolean function g_f corresponding to the pooling layer f performing the $\widetilde{\text{QEC}}_f$ procedure can be expressed as a logic circuit with a constant depth in system size, see Figs. 2 and 4. As a result, the Boolean function G corresponding to the QCNN with d pooling layers can be implemented as a logic circuit with a depth proportional to

d, which can be at most $d = \lfloor \log_3 N \rfloor$ logarithmic in system size N.

We now describe QCNNs detecting the $ZXXXZ$ phase. The QCNN consisting of alternating layers correcting X errors and Z errors is depicted in Fig. 13(a) below. The disentangling unitary $\tilde{U}^\dagger$ consists of controlled Z gates CZ between all neighboring qubits controlled in the computational basis, controlled Y gates $C_y Y$ between all neighboring qubits controlled in the Y basis and Z gates. The X-error correcting procedure $\widetilde{\text{QEC}}_f^X$ consists of controlled Y gates $C_x Y$ and controlled-controlled Y gates $C_x C_x Y$ with controls in the X basis. The Z-error correcting procedure $\widetilde{\text{QEC}}_f^Z$ is the same as for the ZXZ phase involving SWAP, $C_x Z$, and $C_x C_x Z$ gates. The QCNN consisting of alternating layers correcting C errors and Z errors is depicted in Fig. 13(b) below. The C-error correcting procedure $\widetilde{\text{QEC}}_f^C$ consists of $C_x Y$ and $C_x C_x Y$ gates.

As all $\widetilde{\text{QEC}}_f$ procedures consist of gates controlled in the X basis implementing the Pauli Y operation or the Pauli Z operation on the target qubit, they satisfy the condition (6) and thus they can be implemented in classical postprocessing as a Boolean function $g_f(x) = x'$. The output

$$\langle X_j \rangle = \text{Tr}[X_j Q_d \tilde{U}_N^\dagger \rho \tilde{U}_N Q_d^\dagger] = \sum_x \tilde{P}_x (1 - 2G(x)_j), \quad \text{(B3)}$$

of qubit j measured in the fully connected layer of the full quantum QCNN circuit can be determined from bit strings x measured after the first convolutional layer $\tilde{U}_N^\dagger$ by using the jth element of the output of the Boolean function $G(x)$, where $\tilde{P}_x = \text{Tr}[|x\rangle\langle x| \tilde{U}_N^\dagger \rho \tilde{U}_N]$.

Note that the Boolean function corresponding to the C-error correcting procedure for the $ZXXXZ$ phase is the same

as the Boolean function performing the X-error correcting procedure for the ZXZ phase depicted in Fig. 2.

APPENDIX C: PROPAGATION OF SYMMETRY-PRESERVING ERRORS IN QCNN CIRCUITS

In this Appendix, we discuss the propagation of symmetry-preserving X errors in the QCNN detecting the ZXZ phase and consisting of X-error correcting layers. We exploit that the QCNN can be implemented as the constant-depth quantum circuit $U_N^\dagger$, measurements in the X basis and classical postprocessing. We focus on the propagation of errors in the constant-depth quantum circuit $U_N^\dagger$ and the logic circuit implemented in classical postprocessing.

The ZXZ cluster state $|C\rangle$ is mapped by the disentangling unitary $U_N^\dagger$ onto the product state $|+\rangle^{\otimes N}$, where $|+\rangle$ is the $+1$ eigenstate of the Pauli X operator. The subsequent measurement thus deterministically yields the outcome $x_j = 0$ corresponding to $X_j = +1$ for all qubits j. A single X_j error perturbing the cluster state $X_j |C\rangle$ is mapped onto $U_N^\dagger X_j U_N = Z_{j-1} X_j Z_{j+1}$ by the disentangling unitary $U_N^\dagger$ leading to the flip of two measurement outcomes $x_{j\pm1} = 1$, see red lines in Fig. 2. This X-error syndrome is corrected by the X-error correcting procedure such that $g(x)_k = 0$ for all $N/3$ classical bits k propagating to the next layer, see Fig. 2. The other $2 \times N/3$ bits are discarded.

We now investigate the cluster state $K |C\rangle$ perturbed by an arbitrary number $\Upsilon = \sum_{j=1}^N \upsilon_j^{(0)}$ of X errors as the input state, where $K = \prod_{j=1}^N X_j^{\upsilon_j^{(0)}}$ and $\upsilon_j^{(0)} = 0, 1$. The perturbed cluster state is mapped by the disentangling unitary $U_N^\dagger$ onto the product state $\bigotimes_{j=1}^N |\pm\rangle_j$, where qubit j is in the state $|+\rangle$ if $\upsilon_{j-1}^{(0)} \oplus \upsilon_{j+1}^{(0)} = 0$ and in the state $|-\rangle$ otherwise. The subsequent measurement in the Pauli X basis thus yields the deterministic outcome $x_j = \upsilon_{j-1}^{(0)} \oplus \upsilon_{j+1}^{(0)}$. The measured bit string x is processed in the logic circuit depicted in Fig. 2 consisting of X-error correcting layers. A bit x_j at the output of each X-error correcting layer f depends on five bits $x_{j-4\times3^{f-1}}$, $x_{j-2\times3^{f-1}}$, x_j, $x_{j+2\times3^{f-1}}$, and $x_{j+4\times3^{f-1}}$ at the input of this layer. We express the bit string $x_j = \upsilon_{j-3^f}^{(f)} \oplus \upsilon_{j+3^f}^{(f)}$ at every layer $f > 0$ in terms of $\upsilon_j^{(f)} = 0, 1$ similarly to the bit string $x_j = \upsilon_{j-1}^{(0)} \oplus \upsilon_{j+1}^{(0)}$ at layer $f = 0$, i.e., directly after the measurement. We can interpret $\upsilon_j^{(f)}$ as the syndrome of the X_j error on the sublattice with $N/3^f$ qubits. Each X-error correcting layer implements the majority function

$$\upsilon_j^{(f)} = M\left(\upsilon_{j-2\times3^{f-1}}^{(f-1)}, \upsilon_j^{(f-1)}, \upsilon_{j+2\times3^{f-1}}^{(f-1)}\right) \qquad (C1)$$

for the X-error syndromes. As a result, all X-error syndromes are corrected for a small initial number $\Upsilon \ll N$ of X errors and a sufficiently large depth d. The QCNN output thus converges to unity with the increasing depth d.

As shown in Fig. 3, the QCNN output converges to unity for all states in the ZXZ phase. The phase boundary coincides with a threshold density of coherent X errors perturbing the cluster state [14,41]. Above the threshold density, X-error syndromes are concentrated in the QCNN circuit and the QCNN output vanishes with increasing depth d.

We now discuss the propagation of incoherent X errors in the QCNN for the cluster state $\mathcal{E}(|C\rangle\langle C|)$ with the probability p_X of X errors described by the error channel (4) with $p_Y = p_Z = 0$. At layer $f = 0$, the probability $p_0 = p_X$ of the X error is uniform at every qubit and the probabilities of X errors at different qubits j and k are not correlated. We will now describe the probability p_f of X-error syndromes $\upsilon_j^{(f)}$ at each layer f. The majority function (C1) dictates that the probabilities of error syndromes at different qubits remain uniform and uncorrelated. According to the majority function, the error probabilities follow the recursion relation

$$p_f = p_{f-1}^2 (3 - 2p_{f-1}). \qquad (C2)$$

After d X-error correcting layers, we obtain $x_j = \upsilon_{j-3^d}^{(d)} \oplus \upsilon_{j+3^d}^{(d)} = 1$ with the probability $2p_d(1 - p_d)$.

We can exactly determine the QCNN output

$$y = 1 - 4p_d(1 - p_d) \qquad (C3)$$

for the initial error probability p_0 by iterating the recursion relation (C2). The probability p_f after each X-error correcting layer decreases, i.e., $p_f < p_{f-1}$, for $0 < p_{f-1} < 0.5$ and it increases, i.e., $p_f > p_{f-1}$, for $0.5 < p_{f-1} < 1$. As a result, all X-error syndromes are corrected for $p_0 < 0.5$ and a sufficiently large depth d. The QCNN output converges to unity with increasing depth. For $p_0 > 0.5$, the error probability p_d monotonously increases with increasing depth d. For large depths d, we obtain $p_d \approx 1$ and thus the QCNN also attains a near unity output. The situation is qualitatively different for $p_0 = 0.5$. This value of the error probability is the fixed point $p_f = p_{f-1} = 0.5$ of the recursion relation (C2) and the QCNN output vanishes for every depth d.

The action of the error channel $\mathcal{E}$ on the cluster state is qualitatively different for $p_0 = p_X \neq 0.5$ and $p_X = 0.5$. For $p_X \neq 0.5$, the error channel is invertible and it thus preserves SPT order [43]. In the presence of incoherent errors, the SOPs S_{jk} decrease by the factor $(1 - 2p_X)^2$ but they retain nonvanishing values, see Eq. (8). In this case, the QCNN is able to correct the incoherent errors and it attains ideal noise-free values for a sufficiently large depth d. In contrast, the error channel is not invertible for $p_X = 0.5$. The SOPs are annihilated by the error channel [see Eq. (8)] and SPT order is completely washed out [43]. In this case, the QCNN is not able to correct the incoherent errors and it attains a vanishing output for every depth d as discussed above.

APPENDIX D: PROPAGATION OF SYMMETRY-BREAKING ERRORS IN QCNN CIRCUITS

In this Appendix, we discuss the propagation of symmetry-breaking Z errors in QCNNs detecting the ZXZ phase.

The situation is more complicated for Z errors than for symmetry-preserving X errors discussed in Appendix C. A single Z_j error perturbing the cluster state $Z_j |C\rangle$ leads to the flip of a single measurement outcome $x_j = 1$ as the error Z_j commutes with the disentangling unitary $U_N^\dagger$, see Fig. 2. This syndrome of the Z_j error propagates through the X-error correcting layer such that $g(x)_k = 1$ for bits k on the sublattice with $N/3$ bits in the next layer if $k = j - 2$, $k = j$, or $k = j + 2$. As $2 \times N/3$ bits are discarded in each

layer, the density of error syndromes increases. This leads to a decreasing QCNN output for any probability $p_Z \neq 0$ or $p_Z \neq 1$ of Z errors. To correct Z errors, we construct a new Z-error correcting $\widetilde{\mathrm{QEC}}_f^Z$ procedure, depicted in Fig. 10(c), with a corresponding logic circuit depicted in Fig. 4. This logic circuit consists of the majority function, see Eq. (10). The majority function $M(x_{j-7\times3^{f-1}}, x_j, x_{j+7\times3^{f-1}})$ in layer f returns the value of the majority of the three bits $x_{j-7\times3^{f-1}}$, x_j, and $x_{j+7\times3^{f-1}}$. It thus removes isolated syndromes $x_j = 1$ of Z_j errors, see Fig. 4. Provided that the initial density of Z_j errors is small enough, the majority vote further decreases the density of error syndromes, preventing their concentration in the QCNN circuit.

We now investigate the propagation of Z errors in the QCNN with alternating X-error and Z-error correcting layers for the cluster state $\rho = \mathcal{E}(|C\rangle \langle C|)$ with the probability p_Z of Z errors described by the error channel (4) with $p_X = p_Y = 0$. As Z errors commute with the disentangling unitary $U_N^\dagger$, we measure $x_j = 1$ with the identical probability p_Z at all qubits j. Moreover, the probabilities of measuring the values $x_j = 1$ and $x_k = 1$ on different qubits j and k are not correlated. We will now describe how the probability p_f of Z-error syndromes evolves in each layer f of the QCNN. We first note that the probability p_f remains identical in each layer, i.e., the same for all qubits j, as the QCNN circuit is translationally invariant. We also assume that the probability of Z-error syndromes on different qubits remains uncorrelated. This assumption is well justified by the agreement with our numerical simulations showing that correlations between error syndromes that build up in the logic circuit are negligible.

We start with the probability $p_0 = p_Z$ of measuring $x_j = 1$ at each qubit j after the disentangling unitary $U_N^\dagger$. The measured bit strings x are now processed in the X-error correcting layers for f odd and in the Z-error correcting layers for f even. A bit x_j at the output of the X-error correcting layer f depends on five bits $x_{j-4\times3^{f-1}}, x_{j-2\times3^{f-1}}, x_j, x_{j+2\times3^{f-1}}$, and $x_{j+4\times3^{f-1}}$ at the input of this layer, see Fig. 4, each of which has the value 1 with the probability p_{f-1}. Using a truth table for the output of the X-error correcting layer, we determine that the output value $x_j = 1$ occurs with the probability

$$p_f = f_X(p_{f-1})$$
$$= p_{f-1}^3 + p_{f-1}(1 - p_{f-1})^2 (3 - 2p_{f-1} + 4p_{f-1}^2). \quad \text{(D1)}$$

The probability p_f after each X-error correcting layer increases, i.e., $p_f > p_{f-1}$ for $0 < p_{f-1} < 0.5$ resulting in a decreased QCNN output. This can also be seen in Fig. 5, where after each X-error correcting layer, the QCNN output decreases, compare red and blue lines.

A bit x_j at the output of the Z-error correcting layer f depends on three bits $x_{j-7\times3^{f-1}}, x_j$, and $x_{j+7\times3^{f-1}}$ at the input of this layer, see Fig. 4, each of which has the value 1 with the probability p_{f-1}. Using a truth table for the output of the Z-error correcting layer, we determine that the output value $x_j = 1$ occurs with the probability

$$p_f = f_Z(p_{f-1}) = p_{f-1}^2 (3 - 2p_{f-1}). \quad \text{(D2)}$$

The probability p_f after each Z-error correcting layer decreases, i.e., $p_f < p_{f-1}$ for $0 < p_{f-1} < 0.5$ resulting in an increased QCNN output. This can also be seen in Fig. 5,
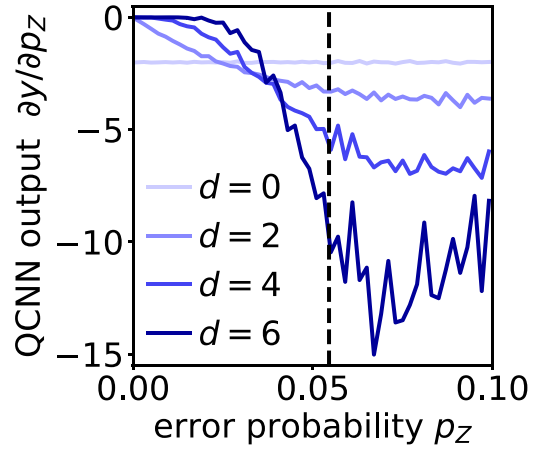

FIG. 11. QCNN consisting of alternating layers correcting X errors and Z errors for the ZXZ cluster state perturbed by symmetry-breaking Z errors. The slope $\partial y / \partial p_z$ of the QCNN output with respect to the probability p_Z of Z errors as a function of the probability p_Z for different depths d. The black-dashed line shows the threshold error probability $p_{\mathrm{th}} = 0.054$. (Parameters: $N = 1215$, $M_S = 10^5$, $p_X = p_Y = 0$).

where after each Z-error correcting layer the QCNN output increases, compare red and blue lines.

We identify two distinct regimes depending on the initial error probability $p_0 = p_Z$. For error probabilities below the threshold $p_Z < p_{\mathrm{th}}$, error correction in even (Z-error correcting) layers dominates over error concentration in odd (X-error correcting) layers resulting in a net reduction of errors after two subsequent layers. For error probabilities above the threshold $p_Z > p_{\mathrm{th}}$, error concentration in odd layers dominates over error correction in even layers resulting in a net concentration of errors after two subsequent layers. We determine the threshold probability $p_{\mathrm{th}} = 0.054$ as the fixed point of the recursion relation $p_f = f_Z(f_X(p_{f-2}))$.

We now study the slope of the QCNN output with respect to the error probability p_Z. We plot the slope as a function of the probability p_Z in Fig. 11 for different depths d of the QCNN. We can see a key difference compared to the slope of the QCNN output with respect to the Hamiltonian parameter h_2 depicted in Fig. 8(b). The slope with respect to h_2 exhibits a sharp dip precisely located at the phase boundary between the ZXZ SPT phase and the paramagnetic phase. In contrast, the slope with respect to the probability p_Z keeps decreasing with the increasing probability p_Z above the threshold value $p_{\mathrm{th}} = 0.054$. We conclude that the sharp dip in the slope of the QCNN output with respect to the Hamiltonian parameters is a unique feature of phase boundaries, which distinguishes them from the behavior of the QCNN output at the threshold error probability p_{th}.

APPENDIX E: QCNN CONSISTING OF X-ERROR CORRECTING LAYERS

In this Appendix, we discuss the QCNN recognizing the ZXZ SPT phase consisting of X-error correcting layers and how it is affected by symmetry-breaking errors. The compact implementation of the QCNN as a quantum circuit consisting
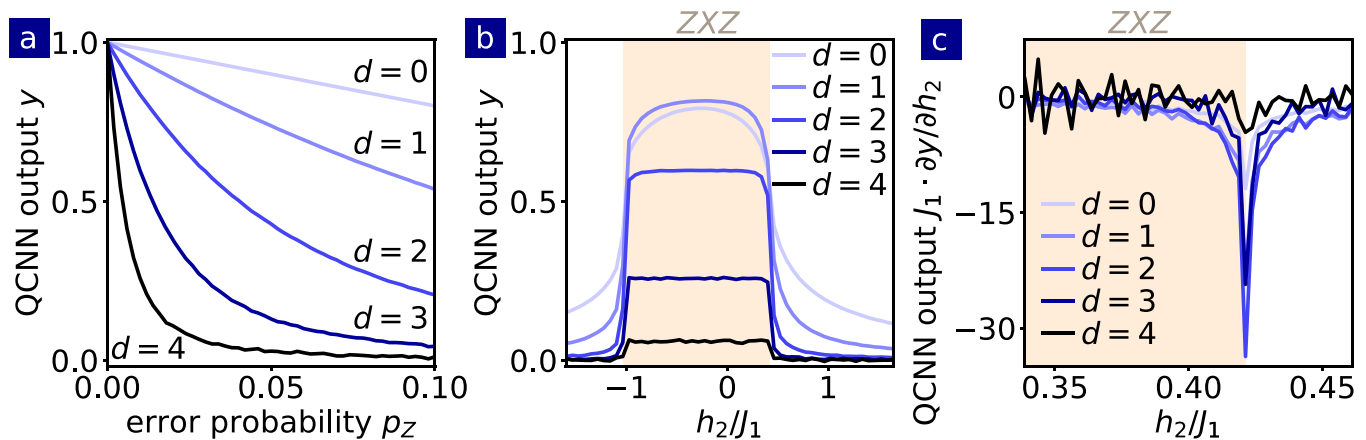
FIG. 12. QCNN detecting the ZXZ SPT phase consisting of X-error correcting layers for the ground states of the cluster-Ising Hamiltonian (1) perturbed by incoherent symmetry-breaking errors. (a) The QCNN output for the ZXZ cluster state perturbed by Z errors as a function of the error probability p_Z for different depths d of the QCNN. (b) The QCNN output as a function of h_2/J_1 and different depths d in the presence of depolarizing noise. (c) The slope of the QCNN output $\partial y/\partial h_2$ with respect to the Hamiltonian parameter h_2 close to the phase boundary between the ZXZ phase and the paramagnetic phase as a function of h_2/J_1 for different depths d. The orange regions denote the ZXZ phase. [Parameters: $N = 1215$, $M_S = 10^4$; (a) $p_X = p_Y = 0$; (b) and (c) $h_1/J_1 = 0.5$, $J_2 = 0$, $p_X = p_Y = p_Z = 0.02$].

of the disentangling unitary $U_N^\dagger$, the measurement of all qubits in the X basis and classical postprocessing is depicted in Fig. 2.

We start by investigating the QCNN for the ZXZ cluster state perturbed by incoherent Z errors as the input state. We plot in Fig. 12(a) the QCNN output as a function of the Z-error probability p_Z. We can see that the QCNN output decreases with increasing depth for all error probabilities. Z errors propagate through the X-error correcting layers and, as the system size is reduced by a factor of three in each layer, the density of Z errors increases. As a result, the probability of Z errors increases with increasing depth d according to the recursion relation (D1). This error concentration leads to the decrease of the QCNN output for any initial error probability p_Z. This is in contrast to the QCNN consisting of alternating layers correcting X errors and Z errors that tolerates symmetry-breaking Z errors below the threshold error probability $p_{\text{th}} = 0.054$.

We now study the QCNN consisting of X-error correcting layers for different ground states of the cluster-Ising Hamiltonian in the presence of depolarizing noise. We plot in Fig. 12(b) the QCNN output as a function of h_2/J_1 and different depths d of the QCNN. We can see that the QCNN output vanishes with increasing depth d both in the SPT phase (orange region) and outside of the SPT phase (white regions). This is in contrast to the QCNN consisting of alternating layers correcting X errors and Z errors that tolerates symmetry-breaking errors and thus attains near unity values in the SPT phase for large depths d.

We plot in Fig. 12(c) the slope $\partial y/\partial h_2$ of the QCNN output y with respect to the Hamiltonian parameter h_2 for different depths d of the QCNN close to the phase boundary between the ZXZ phase and the paramagnetic phase. We can see a dip in the slope of the QCNN output located at the phase boundary. However, the dip vanishes for large depths $d \geqslant 4$ of the QCNN. This is again in contrast to the QCNN consisting of alternating layers correcting X errors and Z errors. For this

network, the dip in the slope becomes more pronounced with increasing depth due to the tolerance of errors.

We conclude that the QCNN consisting of X-error correcting layers is largely affected by symmetry-breaking errors. Due to the concentration of symmetry-breaking errors in the QCNN, the QCNN output rapidly vanishes with the increasing depth d. Also the slope of the QCNN output at phase boundaries vanishes with the increasing depth. This is in contrast to the QCNN consisting of alternating layers correcting X errors and Z errors that tolerates symmetry-breaking errors and thus attains near unity values in the SPT phase for large depths. Moreover, the dip in the slope of the QCNN output at phase boundaries becomes more pronounced with increasing depth.

APPENDIX F: $ZXXXZ$ SPT PHASE

In this Appendix, we discuss QCNNs recognizing the $ZXXXZ$ SPT phase and their tolerance to different types of errors. Similarly, as for the ZXZ SPT phase, we construct a QCNN to recognize the $ZXXXZ$ SPT phase from the paramagnetic phase and the antiferromagnetic phase. The X-error correcting procedure $\widetilde{\text{QEC}}_f^X$ is depicted in Fig. 13(a).

The $ZXXXZ$ cluster state $|D\rangle$ is mapped by the disentangling unitary $\tilde{U}_N^\dagger$ in the first convolutional layer onto the product state $|+\rangle^{\otimes N}$. The subsequent measurement thus deterministically yields the outcome $x_j = 0$ for all qubits j. A single X_j error perturbing the cluster state $X_j|D\rangle$ is mapped onto $\tilde{U}_N^\dagger X_j \tilde{U}_N = Y_{j-2}X_{j-1}X_{j+1}Y_{j+2}$ by the disentangling unitary $\tilde{U}_N^\dagger$ leading to the flip of two measurement outcomes $x_{j\pm2} = 1$. This error is corrected by the X-error correcting QEC procedure such that $g(x)_k = 0$ for all $N/3$ classical bits k propagating to the next layer. Similarly, the syndrome $x_{j\pm2} = x_{j-1} = x_{j+3} = 1$ of a X_jX_{j+1} error is corrected by the X-error correcting procedure.

We now investigate the QCNN with X-error correcting layers. We plot in Fig. 14(a) the QCNN output across a cut through the phase diagram as a function of h_2/J_2 for
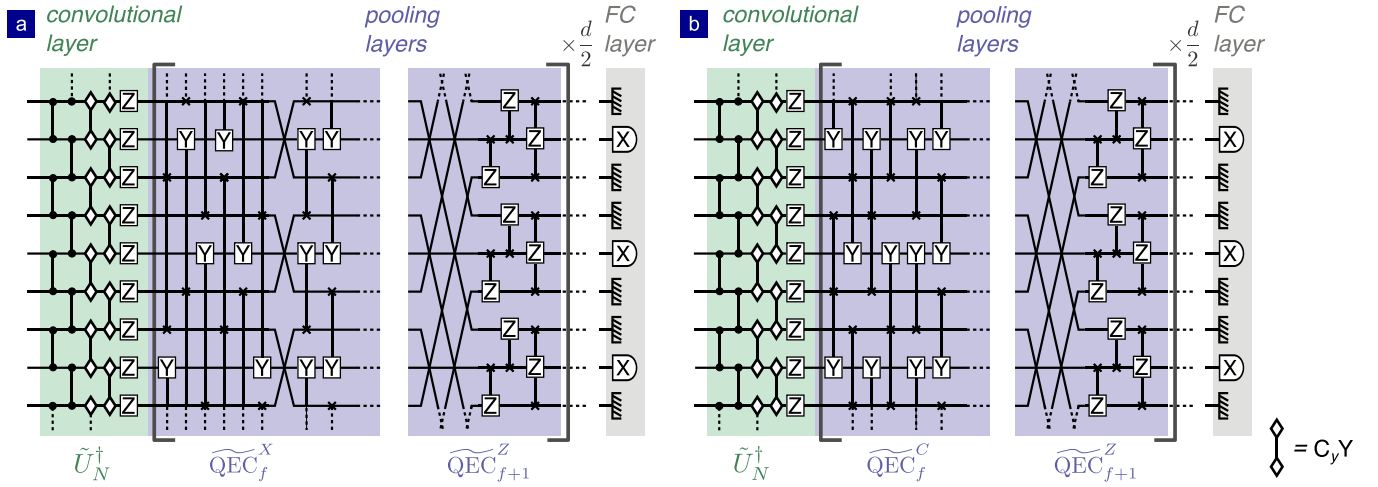
FIG. 13. QCNN quantum circuit detecting the $ZXXXZ$ phase. (a) QCNN with alternating procedures $\widetilde{\mathrm{QEC}}_f^X$ and $\widetilde{\mathrm{QEC}}_{f+1}^Z$ correcting X errors and Z errors, respectively, for the recognition of the $ZXXXZ$ phase from topologically trivial phases. (b) QCNN with alternating procedures $\widetilde{\mathrm{QEC}}_f^C$ and $\widetilde{\mathrm{QEC}}_{f+1}^Z$ correcting C errors and Z errors, respectively, for the recognition of the $ZXXXZ$ phase from the ZXZ phase. The disentangling unitary $\tilde{U}^\dagger$ involves controlled Z gates CZ with controls in the computational basis, controlled Y gates C_yY with controls in the Y basis and Z gates. The $\widetilde{\mathrm{QEC}}_f^X$ procedure consists of SWAP gates as well as controlled-controlled Y gates C_xC_xY and controlled Y gates C_xY with all controls in the X basis. The $\widetilde{\mathrm{QEC}}_{f+1}^Z$ procedure consists of SWAP, C_xC_xZ, and C_xZ gates. The $\widetilde{\mathrm{QEC}}_f^C$ procedure consists of C_xC_xY and C_xY gates.

$J_1 = 0$ and different depths d of the QCNN in the presence of incoherent X errors. We can see that the QCNN output converges to unity with the increasing depth of the QCNN in the $ZXXXZ$ SPT phase (gray region). On the other hand, the QCNN output vanishes with the increasing depth in the topologically trivial phases (white regions). This shows that the QCNN can recognize the $ZXXXZ$ SPT phase from topologically trivial phases in the presence of incoherent X errors. Incoherent X errors can be tolerated for any probability $p_X \neq 0.5$.

To equip the QCNN with the tolerance to symmetry-breaking Z errors, we alternate X-error correcting layers with Z-error correcting layers, see Fig. 13(a). The $\widetilde{\mathrm{QEC}}_f^Z$ procedure, correcting Z errors, is the same as that for the ZXZ phase, cf. Fig. 10(c), which can be implemented in classical postprocessing as the majority function (10), depicted in Fig. 4. We start by investigating the QCNN with alternating layers for the $ZXXXZ$ cluster state perturbed by incoherent Z errors as the input state. We plot in Fig. 14(b) the QCNN output as a function of the Z-error probability. We can see
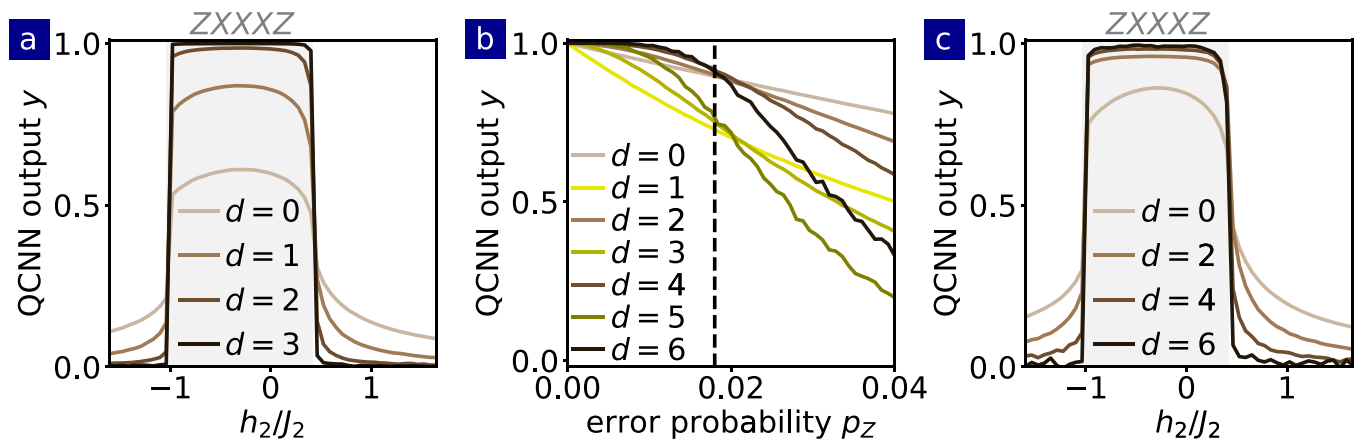

FIG. 14. QCNN recognizing the $ZXXXZ$ phase from the paramagnetic phase and the antiferromagnetic phase. (a) The output of the QCNN consisting of X-error correcting layers for ground states of the cluster-Ising Hamiltonian (1) perturbed by incoherent X errors as a function of h_2/J_2 for different depths d of the QCNN. (b) The output of the QCNN consisting of alternating layers correcting X errors and Z errors as a function of the Z-error probability p_Z for the $ZXXXZ$ cluster state for different depths d. The black dashed line shows the threshold error probability $\tilde{p}_{th} = 0.018$. (c) The output of the QCNN consisting of alternating layers correcting X errors and Z errors for ground states of the cluster-Ising Hamiltonian (1) perturbed by depolarizing noise as a function of h_2/J_2 for different depths d. The gray regions denote the $ZXXXZ$ phase. [Parameters: $N = 1215$, $M_S = 10^4$; (a) $p_X = 0.1$, $p_Y = p_Z = 0$, $h_1/J_2 = 0.5$, $J_1 = 0$; (b) $p_X = p_Y = 0$; (c) $p_X = p_Y = p_Z = 0.005$, $h_1/J_2 = 0.5$, $J_1 = 0$].

an alternating QCNN output after odd and even layers. Z errors propagate through odd X-error correcting layers and, as the system size is reduced by a factor of three, the density of Z errors increases. This error concentration leads to the decrease of the QCNN output after odd layers, compare brown and yellow lines in Fig. 14(b). In contrast, even layers correct Z errors leading to the decrease of their density and the increase of the QCNN output. The QCNN can tolerate Z errors below the threshold error probability $\tilde{p}_{\text{th}} = 0.018$ as the error correction in even layers dominates over the error concentration in odd layers. This shows that implementing the Z-error correcting layer after each X-error correcting layer prevents the concentration of symmetry-breaking Z errors for small error probabilities.

The threshold probability $\tilde{p}_{\text{th}} = 0.018$ for the $ZXXXZ$ phase is smaller than the threshold probability $p_{\text{th}} = 0.054$ for the ZXZ phase. This decrease in the tolerated error probabilities can be understood by investigating the propagation of Z errors in the QCNN circuit. In contrast to the disentangling circuit $U_N^\dagger$ for the ZXZ phase, which commutes with Z_j errors, the disentangling unitary $\tilde{U}^\dagger$ for the $ZXXXZ$ phase maps Z_j errors onto three errors $\tilde{U}_N^\dagger Z_j \tilde{U}_N = Y_{j-1} Z_j Y_{j+1}$, which flip the measurement outcomes $x_j = x_{j\pm1} = 1$ on qubits $j-1$, j and $j+1$. This error syndrome is corrected by the Z-error correcting procedure provided that the Z_j error is isolated. However, the threshold error probability $\tilde{p}_{\text{th}}$ is smaller than for the ZXZ phase due to the multiplication of symmetry-breaking errors by the disentangling unitary.

We now study the error tolerance of the QCNN with alternating layers for different ground states of the cluster-Ising Hamiltonian (1) perturbed by depolarizing noise. We plot in Fig. 14(c) the QCNN output as a function of h_2/J_2 for $J_1 = 0$ and different depths d of the QCNN. We can see that the QCNN tolerates the incoherent errors due to depolarizing noise as its output converges to unity with the increasing depth d in the $ZXXXZ$ phase (gray region) and vanishes in the topologically trivial phases (white regions).

In conclusion, we constructed a QCNN for the $ZXXXZ$ phase that tolerates symmetry-preserving X errors if the error channel is invertible and symmetry-breaking Z errors for small error probabilities. The QCNN is constructed similarly as for the ZXZ phase by amending the QEC procedures to correct X_j and $X_j X_{j+1}$ errors perturbing the $ZXXXZ$ cluster state. As the disentangling unitary $\tilde{U}_N^\dagger$ for the $ZXXXZ$ phase maps Z_j errors onto three errors, $\tilde{U}_N^\dagger Z_j \tilde{U}_N = Y_{j-1} Z_j Y_{j+1}$, the threshold probability $\tilde{p}_{\text{th}} = 0.018$ of Z errors is reduced compared to the QCNN for the ZXZ phase.

We finally discuss the propagation of errors in the QCNN detecting the $ZXXXZ$ phase from the ZXZ phase. This QCNN consists of alternating C-error and Z-error correcting layers as depicted in Fig. 13(b) and discussed in Sec. VI of the main text. The C-error correcting layers are essential for the detection of the $ZXXXZ$ phase while the Z-error correcting layers equip the QCNN with error tolerance. A single C_j error is transformed by the disentangling unitary $\tilde{U}_N^\dagger$ as $\tilde{U}_N^\dagger C_j \tilde{U}_N \propto Y_{j-1} X_j Y_{j+1}$ and thus leads to the error syndrome $x_{j\pm1} = 1$. This error syndrome is corrected by the C-error correcting procedure. X_j and $X_j X_{j+1}$ errors are mapped onto $\tilde{U}_N^\dagger X_j \tilde{U}_N = Y_{j-2} X_{j-1} X_j X_{j+1} Y_{j+2}$ and $\tilde{U}_N^\dagger X_j X_{j+1} \tilde{U}_N = Y_{j-2} Z_{j-1} Z_{j+2} Y_{j+3}$

with the corresponding error syndromes $x_{j\pm2} = 1$ and $x_{j\pm2} = x_{j-1} = x_{j+3} = 1$, respectively. The syndrome $x_{j+2} = 1$ of the X_j error is corrected by the C-error correcting layer only if bit j propagates to the next layer. If bit j is discarded, the X_j-error syndrome is transformed into either $x_{j-2} = x_{j+4} = 1$ or into $x_{j-4} = x_{j+2} = 1$. On the sublattice with $N/3$ bits in the next layer, this corresponds in both cases to the C_k-error syndrome $x_{k\pm3} = 1$. In the subsequent Z-error correcting layer, we take the majority value $M(x_{k-7\times3}, x_k, x_{k+7\times3})$ of every triple of qubits $x_{k-7\times3}$, x_k, and $x_{k+7\times3}$. As these bits are separated by the distance 7×3, the single C_k-error syndrome does not change any of the majority values and it is thus removed. Similarly, also $X_j X_{j+1}$-error syndromes are removed by two subsequent layers. The QCNN can thus distinguish the $ZXXXZ$ phase from the ZXZ phase, see Fig. 7.

Note that the QCNN consisting of only C-error correcting layers also corrects X_j and $X_j X_{j+1}$ errors. As we discussed above, the syndrome of the X_j error is transformed by the C-error correcting layer into the C_k-error syndrome on the sublattice with $N/3$ bits. This C_k-error syndrome is corrected in the subsequent C-error correcting layer. Similarly, the syndrome of a single $X_j X_{j+1}$ error is also corrected by two subsequent C-error correcting layers.

APPENDIX G: MULTISCALE STRING ORDER PARAMETER

In this Appendix, we describe the multiscale SOP S_{M}, see Eq. (11), that is measured by the QCNNs considered in this work. First, we show that S_{M} is a sum of products of SOPs S_{jk}. Then, we demonstrate that the length of the SOPs involved in S_{M} increases exponentially with the depth d of the QCNN. Finally, we determine a lower bound for the number of products of SOPs that are summed together to construct S_{M}.

We focus here on the QCNN detecting the ZXZ phase, consisting of alternating X-error and Z-error correcting layers. We consider the form of the QCNN depicted in Fig. 10(c) with all convolutional layers for $f > 1$ and the fully connected layer absorbed into the $\widetilde{\text{QEC}}_f$ procedures in pooling layers. The QCNN circuit thus performs the unitary

$$U_{\text{QCNN}} = \widetilde{\text{QEC}}_d^X \widetilde{\text{QEC}}_{d-1}^Z ... \widetilde{\text{QEC}}_2^Z \widetilde{\text{QEC}}_1^X U_N^\dagger \qquad \text{(G1)}$$

consisting of the disentangling unitary $U_N^\dagger$ and d pooling layers $\widetilde{\text{QEC}}_f$ where $f = 1, 2, ..., d$. For odd (even) f, the pooling layers perform the X-error (Z-error) correcting procedure $\widetilde{\text{QEC}}_f^X$ ($\widetilde{\text{QEC}}_f^Z$), see Fig. 10(c). We also assume that the QCNN has an odd number d of layers.

Sum of products of string order parameters. We first show that the observable measured by the QCNN corresponds to the multiscale SOP S_{M}, which is a sum of products of SOPs, cf. Eq. (11).

The measurement of the Pauli operator

$$\langle X_{\frac{N+1}{2}} \rangle = \text{Tr}[X_{\frac{N+1}{2}} U_{\text{QCNN}} \rho U_{\text{QCNN}}^\dagger]$$
$$= \text{Tr}[U_{\text{QCNN}}^\dagger X_{\frac{N+1}{2}} U_{\text{QCNN}} \rho] \qquad \text{(G2)}$$

at the end of the QCNN circuit corresponds to the measurement of the observable $U_{\text{QCNN}}^\dagger X_{\frac{N+1}{2}} U_{\text{QCNN}}$ on the input

TABLE II. Backpropagation of the measured observable through the QCNN circuit with the depth d. The table displays the measured observable backpropagated to layer f, where the ellipsis denotes products of two and more products of Pauli strings. The table also displays the number of products of Pauli strings $G_{jk}^{(f)}$ involved in the measured observable and the unitary performed at layer f. The multiscale SOP S_M measured on the input state and the number of products of SOPs involved in S_M are stated in the last row. We use $H_{jk}^{(d-2)} = G_{jk}^{(d-2)} G_{(j+3^{d-2})(k-3^{d-2})}^{(d-2)}$, $i = \frac{N+1}{2}$, $\alpha', \beta' \in \{0, 2 \times 3^{d-1}, 4 \times 3^{d-1}\}$, $\gamma' = 4 \times 3^{d-1}$, $\kappa \in \{-7 \times 3^{d-2}, 0, 7 \times 3^{d-2}\}$, $\lambda \in \{-6 \times 3^{d-2}, 6 \times 3^{d-2}\}$, and $\lambda' \in \{-6 \times 3^{d-2}, 0, 6 \times 3^{d-2}\}$.

f	Measured observable	# of products	
d	$G_{ii}^{(d)} = X_i$	1	$\widetilde{\text{QEC}}_d^X$
$d-1$	$\frac{1}{4} \sum_{\alpha'\beta'} G_{(i-\alpha')(i+\beta')}^{(d-1)} - \frac{1}{4} \sum_{\alpha'} G_{(i-\alpha')i}^{(d-1)} G_{(i+\gamma')(i+\gamma')}^{(d-1)} - \frac{1}{4} \sum_{\alpha'} G_{(i-\gamma')(i-\gamma')}^{(d-1)} G_{i(i+\alpha')}^{(d-1)} \cdots$	16	$\widetilde{\text{QEC}}_{d-1}^Z$
$d-2$	$\frac{1}{8} \sum_{\kappa} G_{(i+\kappa)(i+\kappa)}^{(d-2)} - \frac{1}{16} \sum_{\lambda} G_{(i+\lambda-3^{d-2})(i+\lambda+3^{d-2})}^{(d-2)} + \frac{1}{32} \sum_{\lambda'} H_{(i+\lambda'-3^{d-2})(i+\lambda'+3^{d-2})}^{(d-2)} \cdots$	2500	$\widetilde{\text{QEC}}_{d-2}^X$
	$\vdots$		
0	$\sum_{ij} \eta_{ij}^{(1)} G_{(i+1)(j-1)}^{(0)} + \sum_{ijkl} \eta_{ijkl}^{(2)} G_{(i+1)(j-1)}^{(0)} G_{(k+1)(l-1)}^{(0)} + \cdots$	$\Omega(2^{3^{d-2}})$	$U_N^\dagger$
Input	$S_M = \sum_{ij} \eta_{ij}^{(1)} S_{ij} + \sum_{ijkl} \eta_{ijkl}^{(2)} S_{ij} S_{kl} + \cdots$	$\Omega(2^{3^{d-2}})$	

state ρ. We used the cyclic property of the trace in the second equality in Eq. (G2). We backpropagate the measured observable through the QCNN circuit to the input state (zeroth layer). To this end, we use the recursion relations

$$\widetilde{\text{QEC}}_f^{X\,\dagger} G_{jk}^{(f)} \widetilde{\text{QEC}}_f^X = \frac{1}{4}\left[\sum_{\alpha\beta} G_{(j-\alpha)(k+\beta)}^{(f-1)} - \sum_\alpha G_{(j-\alpha)k}^{(f-1)} G_{(k+\gamma)(k+\gamma)}^{(f-1)} - \sum_\alpha G_{(j-\gamma)(j-\gamma)}^{(f-1)} G_{j(k+\alpha)}^{(f-1)} + G_{(j-\gamma)(j-\gamma)}^{(f-1)} G_{jk}^{(f-1)} G_{(k+\gamma)(k+\gamma)}^{(f-1)} \right],$$

(G3)

$$\widetilde{\text{QEC}}_f^{Z\,\dagger} G_{jk}^{(f)} \widetilde{\text{QEC}}_f^Z = \frac{1}{2^{l_f}} \prod_\delta [X_{\delta-\epsilon} + X_\delta + X_{\delta+\epsilon} - X_{\delta-\epsilon} X_\delta X_{\delta+\epsilon}],$$

(G4)

for f odd and f even, respectively, where $\alpha, \beta \in \{0, 2 \times 3^{f-1}, 4 \times 3^{f-1}\}$, $\gamma = 4 \times 3^{f-1}$, $\delta \in \{j, j + 2 \times 3^f, ..., k\}$ and $\epsilon = 7 \times 3^{f-1}$. These relations describe the backpropagation of Pauli strings

$$G_{jk}^{(f)} = X_j X_{j+2\times3^f}...X_k. \tag{G5}$$

The length of the Pauli strings $G_{jk}^{(f)}$ is defined as $l_f = (k - j)/(2 \times 3^f) + 1$.

The backpropagation of the measured observable is summarized in Table II. The Pauli operator $X_{\frac{N+1}{2}}$ measured at the end of the QCNN circuit corresponds to the Pauli string $G_{\frac{N+1}{2}\frac{N+1}{2}}^{(d)} = X_{\frac{N+1}{2}}$ with the minimal length $l_d = 1$. The recursion relation (G3) dictates that backpropagating this operator through the X-error correcting layer $\widetilde{\text{QEC}}_d^X$ gives rise to a sum of 16 terms including nine Pauli strings $G_{jk}^{(d-1)}$, six products of two Pauli strings $G_{jk}^{(d-1)}$ and a single product of three Pauli strings $G_{jk}^{(d-1)}$ at layer $f = d - 1$, see Table II.

Next, we backpropagate these Pauli strings and the products of Pauli strings through the layer $\widetilde{\text{QEC}}_{d-1}^Z$ performing the Z-error correcting procedure. Due to the linearity of the unitary $\widetilde{\text{QEC}}_{d-1}^Z$, we can separately backpropagate each product in the sum. Each Pauli string $G_{jk}^{(d-1)}$ gives rise to $4^{l_{d-1}}$ products of Pauli X_i operators at layer $f = d - 2$, see Eq. (G4). These products can be expressed in terms of Pauli strings $G_{jk}^{(d-2)}$ by

using Eq. (G5). We thus again obtain a sum of products of Pauli strings $G_{jk}^{(d-2)}$ at layer $f = d - 2$, see Table II.

We continue backpropagating these products of Pauli strings towards the input state at layer $f = 0$. Backpropagating the Pauli string $G_{jk}^{(f)}$ through the X-error correcting layer $\widetilde{\text{QEC}}_f^X$ gives rise to 16 products of Pauli strings $G_{jk}^{(f-1)}$, see Eq. (G3). In every X-error correcting layer as well as in every Z-error correcting layer, we again obtain a sum of products of Pauli strings $G_{jk}^{(f)}$. At layer $f = 0$, Pauli strings $G_{jk}^{(0)}$ are mapped by the disentangling unitary $U_N^\dagger$ onto SOPs,

$$U_N G_{jk}^{(0)} U_N^\dagger = S_{(j-1)(k+1)}, \tag{G6}$$

see Table II. As a result, we measure on the input state a sum of products of SOPs, i.e., the multiscale SOP S_M of Eq. (11).

Length of string order parameters. The backpropagation of all Pauli strings and their products is intractable due to their rapidly increasing number with the depth of the QCNN, see Table II. However, we now show that the multiscale SOP involves a SOP whose length increases exponentially with the depth d of the QCNN.

To this end, we focus on the product

$$H_{jk}^{(f)} = \mathcal{L}_j^{(f)} G_{jk}^{(f)} G_{(j+3^f)(k-3^f)}^{(f)} \mathcal{R}_k^{(f)} \tag{G7}$$

of Pauli strings $G_{jk}^{(f)}$, $G_{(j-3^f)(k+3^f)}^{(f)}$, $\mathcal{L}_j^{(f)}$ and $\mathcal{R}_k^{(f)}$. The Pauli strings $\mathcal{L}_j^{(d-2)} = \mathcal{R}_k^{(d-2)} = \mathbb{1}$ reduce to the identity operator

TABLE III. Backpropagation of products $H_{jk}^{(f)}$ of Pauli strings through the QCNN circuit with the depth d. We focus on a single product at each layer $f = 1, 2, ..., d-2$. The table displays the length l_f of the Pauli string $G_{(j+3^f)(k-3^f)}^{(f)}$ in the product $H_{jk}^{(f)}$ and the unitary performed at layer f. The corresponding product of SOPs measured on the input state and the length L of the SOP $S_{(i-l_0)(i+l_0)}$ are displayed in the last row. Operators $\tilde{\mathcal{L}}$ and $\tilde{\mathcal{R}}$ are defined in Appendix I and $i = \frac{N+1}{2}$.

f	Product of Pauli strings	l_f	
$d-2$	$H_{(i-3^{d-2})(i+3^{d-2})}^{(d-2)}$	1	$\widetilde{\text{QEC}}_{d-2}^X$
	$\vdots$		
f even	$H_{(i-l_f \times 3^f)(i+l_f \times 3^f)}^{(f)}$	$\frac{3^{d-f}+13}{8}$	$\widetilde{\text{QEC}}_f^Z$
f odd	$H_{(i-l_f \times 3^f)(i+l_f \times 3^f)}^{(f)}$	$\frac{3^{d-f}-1}{8}$	$\widetilde{\text{QEC}}_f^X$
	$\vdots$		
0	$H_{(i-l_0)(i+l_0)}^{(0)}$	$\frac{3^d+13}{8}$	$U_N^\dagger$
Input	$\tilde{\mathcal{L}} S_{(i-l_0-1)(i+l_0+1)} S_{(i-l_0)(i+l_0)} \tilde{\mathcal{R}}$	$L = \frac{3^d+17}{4}$	

at layer $f = d-2$ and they are defined recursively for $f < d-2$ by relations

$$\mathcal{L}_j^{(f)} = \mathcal{L}_{j+2 \times 3^f}^{(f+1)}, \tag{G8}$$

$$\mathcal{R}_k^{(f)} = \mathcal{R}_{k-2 \times 3^f}^{(f+1)}, \tag{G9}$$

for f being even and

$$\mathcal{L}_j^{(f)} = \mathcal{L}_{j-5 \times 3^f}^{(f+1)} X_{j-6 \times 3^f} X_{j-3 \times 3^f}, \tag{G10}$$

$$\mathcal{R}_k^{(f)} = X_{k+3 \times 3^f} X_{k+6 \times 3^f} \mathcal{R}_{k+5 \times 3^f}^{(f+1)}, \tag{G11}$$

for f being odd.

We show in Appendix I that the product $H_{jk}^{(f)}$ appears at every layer $f \leqslant d-2$. The backpropagation of the products $H_{jk}^{(f)}$ is summarized in Table III. The first product $H_{(\frac{N+1}{2}-3^{d-2})(\frac{N+1}{2}+3^{d-2})}^{(d-2)}$ appears at layer $f = d-2$, see Table II. The product $H_{jk}^{(f)}$ recursively appears at every layer $f < d-2$. After the disentangling unitary $U_N^\dagger$ at layer $f = 0$, the product $H_{jk}^{(0)}$ gives rise to a product of SOPs, see Table III.

The Pauli sting $G_{(\frac{N+1}{2}-l_0+1)(\frac{N+1}{2}+l_0-1)}^{(0)}$ in the product $H_{(\frac{N+1}{2}-l_0)(\frac{N+1}{2}+l_0)}^{(0)}$ at layer $f = 0$ attains the length $l_0 = \frac{3^d+13}{8}$, see Table III. The Pauli string $G_{(\frac{N+1}{2}-l_0+1)(\frac{N+1}{2}+l_0-1)}^{(0)}$ is mapped by the disentangling unitary $U_N^\dagger$ onto the SOP $S_{(\frac{N+1}{2}-l_0)(\frac{N+1}{2}+l_0)}$ with the length

$$L = 2l_0 + 1 = \frac{3^d+17}{4} \sim 3^d. \tag{G12}$$

This shows that the multiscale SOP (11) involves a SOP whose length increases exponentially for large depths d of the QCNN. For the depth $d = \log_3 N$, this SOP exhibits the length $L \approx N/4$ comparable to system size N. By extending the analysis presented here, it can be shown that the multiscale SOP S_M involves also other SOPs with exponentially increasing lengths $L \sim 3^d$ as well as SOPs at all length scales between $L = 1$ and $L \sim 3^d$.

TABLE IV. Backpropagation of products of Pauli strings through the QCNN circuit with the depth d. We focus on a single product of Pauli strings $H_{jk}^{(2)}$ at layers $f = 1, 2$ and display all products of SOPs measured on the input state that emerge by backpropagating the product $H_{jk}^{(2)}$ through $\widetilde{\text{QEC}}_1^X U_N^\dagger$. The table displays the length l_f of the Pauli string $G_{(j+3^f)(k-3^f)}^{(f)}$ in the product $H_{jk}^{(f)}$ and the unitary performed at layer f. $C_\zeta = S_{(\zeta-1)(\zeta+1)}$ are stabilizer elements as defined in the main text, A_ζ and B_ζ are defined in Eqs. (G14) and (G15), respectively, operators $\bar{\mathcal{L}}$ and $\bar{\mathcal{R}}$ are defined in the Appendix I, $\zeta \in \{\frac{N+1}{2} - 9l_2, \frac{N+1}{2} - 9(l_2-1), ..., \frac{N+1}{2} + 9l_2\}$, and $i = \frac{N+1}{2}$.

f	Products of Pauli strings	l_f	
2	$H_{(i-9l_2)(i+9l_2)}^{(2)}$	$\frac{3^{d-2}+13}{8}$	$\widetilde{\text{QEC}}_2^Z$
1	$H_{(i-9l_2)(i+9l_2)}^{(2)}$		$\widetilde{\text{QEC}}_1^X U_N^\dagger$
Input	$\bar{\mathcal{L}}\left(\prod_\zeta A_\zeta C_\zeta B_\zeta \right) \bar{\mathcal{R}}$		

Number of products of string order parameters. Finally, we determine a lower bound for the number of products of SOPs in the multiscale SOP S_M. To this end, we focus on products of Pauli strings displayed in Table IV.

We start with the product $H_{(\frac{N+1}{2}-9l_2)(\frac{N+1}{2}+9l_2)}^{(2)}$, which appears at layer $f = 2$, see Table III. The recursion relation (G4) dictates that this product appears at layer $f = 1$ as well. In contrast to the discussion above, we now focus on the product $H_{(\frac{N+1}{2}-9l_2)(\frac{N+1}{2}+9l_2)}^{(2)}$ at layer $f = 1$. The Pauli strings $G_{[\frac{N+1}{2}-9(l_2-1)][\frac{N+1}{2}+9(l_2-1)]}^{(2)}$ and $G_{(\frac{N+1}{2}-9l_2)(\frac{N+1}{2}+9l_2)}^{(2)}$ in this product have lengths $l_2 = \frac{3^{d-2}+13}{8}$ and $l_2 + 1$. We backpropagate this product through the X-error correcting layer $\widetilde{\text{QEC}}_1^X$ and the disentangling unitary $U_N^\dagger$,

$$U_N \widetilde{\text{QEC}}_1^{X\dagger} H_{(i-9l_2)(i+9l_2)}^{(2)} \widetilde{\text{QEC}}_1^X U_N^\dagger$$

$$= \bar{\mathcal{L}}\left(\prod_\zeta A_\zeta C_\zeta B_\zeta \right) \bar{\mathcal{R}}, \tag{G13}$$

where $\zeta \in \{\frac{N+1}{2} - 9l_2, \frac{N+1}{2} - 9(l_2-1), ..., \frac{N+1}{2} + 9l_2\}$, $C_\zeta = S_{(\zeta-1)(\zeta+1)}$ are stabilizer elements as defined in the main text, and

$$A_\zeta = \frac{1}{2}\left(C_{\zeta-4}C_{\zeta-2} - C_{\zeta-4} + C_{\zeta-2} + \mathbb{1} \right), \tag{G14}$$

$$B_\zeta = \frac{1}{2}\left(\mathbb{1} + C_{\zeta+2} - C_{\zeta+4} + C_{\zeta+2}C_{\zeta+4} \right). \tag{G15}$$

In Eq. (G13), we used the recursion relation (G3) as well as Eq. (G6), see Appendix I for details. The product (G13) involves $2l_2 + 1 = \frac{3^{d-2}+17}{4} > \frac{3^{d-2}}{4}$ terms A_ζ and B_ζ. By distributing the parentheses in all terms A_ζ and B_ζ, we obtain a sum of $16^{2l_2+1} > 2^{3^{d-2}}$ products of SOPs S_{jk}. Note that these products of SOPs emerge from only the single product $H_{(\frac{N+1}{2}-9l_2)(\frac{N+1}{2}+9l_2)}^{(2)}$ at layer $f = 1$. This places the lower bound $2^{3^{d-2}}$ on the total number of products of SOPs in the multiscale SOP S_M, which involves also many other products of SOPs.

In summary, we showed in this Appendix that the QCNN with alternating X-error and Z-error correcting layers detecting the ZXZ phase measures the multiscale SOP S_M, see

Eq. (11). This multiscale SOP is a sum of products of SOPs S_{jk} whose length $L \sim 3^d$ increases exponentially with the depth d of the QCNN. The lower bound for the number of products of SOPs in the sum is $2^{3^{d-2}}$.

APPENDIX H: SAMPLE COMPLEXITY OF THE MULTISCALE STRING ORDER PARAMETER

In this Appendix, we discuss the sample complexity of directly sampling the multiscale SOP of Eq. (11) from the input state via local Pauli measurements without using any quantum circuit.

A local Pauli measurement consists of simultaneously reading out all qubits j in the basis of Pauli operators $\sigma_j = X_j, Y_j, Z_j$. We thus measure in the basis of the tensor product $\mathcal{B} = \bigotimes_{j=1}^{N} \sigma_j$ of the Pauli operators σ_j. A product of SOPs can be sampled via the local Pauli measurement in any Basis $\mathcal{B}$ in which it is diagonal. Several products of SOPs that are diagonal in the same tensor product basis can be simultaneously sampled in this basis. We can express the multiscale SOP

$$S_{\mathrm{M}} = \sum_{m=1}^{b} O_m \tag{H1}$$

in terms of operators O_m, which are sums of products of SOPs that are diagonal in the tensor product basis $\mathcal{B}_m$. To determine the expectation value of the multiscale SOP, we can individually measure each operator O_m using local Pauli measurements. While the sample complexity of this measurement depends on the variance $\langle O_m^2 \rangle - \langle O_m \rangle^2$ of the operators O_m, the number b of different bases $\mathcal{B}_m$ in which we need to measure places a lower bound on the sample complexity. Note that the decomposition (H1) of the multiscale SOP is not unique as a product of SOPs can be diagonal in several bases $\mathcal{B}_m$. To determine the lower bound for the sample complexity of the multiscale SOP, we now investigate the minimal number B of bases in which we need to measure.

To this end, we focus on the products (G13) of SOPs. We rewrite the products (G13) as

$$\bar{\mathcal{L}} \left(\prod_{\zeta} A_{\zeta} C_{\zeta} B_{\zeta} \right) \bar{\mathcal{R}} = \bar{\mathcal{L}} A_{\frac{N+1}{2} - 9l_2} \left(\prod_{\zeta} C_{\zeta} \right)$$
$$\times \left(\prod_{\zeta'} K_{\zeta'} \right) B_{\frac{N+1}{2} + 9l_2} \bar{\mathcal{R}}, \tag{H2}$$

where $\zeta' \in \{ \frac{N+1}{2} - 9l_2 + 4, \frac{N+1}{2} - 9l_2 + 13, ..., \frac{N+1}{2} + 9l_2 - 5 \}$ and

$$K_{\zeta'} = \tfrac{1}{4}[(C_{\zeta'-2} + \mathbb{1})(C_{\zeta'+3} + \mathbb{1}) \tag{H3}$$
$$+ (C_{\zeta'-2} - \mathbb{1})C_{\zeta'}(C_{\zeta'+3} + \mathbb{1}) \tag{H4}$$
$$+ (C_{\zeta'-2} + \mathbb{1})C_{\zeta'+1}(C_{\zeta'+3} - \mathbb{1}) \tag{H5}$$
$$+ (C_{\zeta'-2} - \mathbb{1})C_{\zeta'}C_{\zeta'+1}(C_{\zeta'+3} - \mathbb{1})]. \tag{H6}$$

Crucially, each operator $K_{\zeta'}$ involves terms that need to be measured in three different tensor product bases. Recalling

that $C_{\zeta'} = Z_{\zeta'-1}X_{\zeta'}Z_{\zeta'+1}$, the terms in the line (H4) are diagonal in the X basis on qubit ζ' and in the computational basis on qubit $\zeta' + 1$. The terms in the line (H5) are diagonal in the computational basis on qubit ζ' and in the X basis on qubit $\zeta' + 1$. The terms in the line (H6) are diagonal in the Y basis on qubits ζ' and $\zeta' + 1$. As a result, we need to measure in three different bases $X_{\zeta'}Z_{\zeta'+1}$, $Z_{\zeta'}X_{\zeta'+1}$ and $Y_{\zeta'}Y_{\zeta'+1}$ on qubits ζ' and $\zeta' + 1$. The terms in the line (H3) are diagonal in any of these three bases as they act as the identity operator on qubits ζ' and $\zeta' + 1$.

The product of SOPs (H2) involves $2 l_2 = \frac{3^{d-2}+13}{4} > 3^{d-4}$ operators $K_{\zeta'}$. Distributing the parentheses in all operators $K_{\zeta'}$ on the right-hand side of Eq. (H2) gives rise to products of SOPs with mutually incompatible bases $\bigotimes_{\zeta'} \{X_{\zeta'}Z_{\zeta'+1}, Z_{\zeta'}X_{\zeta'+1}, Y_{\zeta'}Y_{\zeta'+1}\}$ on qubits ζ' and $\zeta' + 1$. As a result, we need to measure in $3^{2l_2} > 3^{3^{d-4}}$ different tensor product bases $\mathcal{B}_m$. This places the lower bound $3^{3^{d-4}}$ for the sample complexity of measuring the multiscale SOP via local Pauli measurements.

APPENDIX I: BACKPROPAGATION OF PRODUCTS (G7) OF PAULI STRINGS THROUGH THE QCNN CIRCUIT

In this Appendix, we discuss the backpropagation of products $H_{jk}^{(f)}$ of Pauli strings defined in Eq. (G7) through the QCNN circuit. We show that the product $H_{jk}^{(f)}$ recursively appears at every layer $f < d - 2$ of the QCNN. We also derive Eq. (G13) describing the backpropagation of the product $H_{jk}^{(2)}$ through the first X-error correcting layer $\widetilde{\mathrm{QEC}}_1^X$ and the disentangling unitary $U_N^\dagger$.

Following the recursion relations (G8)–(G11), the Pauli strings $\mathcal{L}_j^{(f)}$ and $\mathcal{R}_k^{(f)}$ in the product $H_{jk}^{(f)}$ for f being odd can be explicitly expressed as

$$\mathcal{L}_j^{(f)} = \prod_{g=0}^{(d-4-f)/2} X_{j-\kappa_g-3\times 3^{f+2g}}X_{j-\kappa_g}, \tag{I1}$$

$$\mathcal{R}_k^{(f)} = \prod_{g=0}^{(d-4-f)/2} X_{k+\kappa_g}X_{k+\kappa_g+3\times 3^{f+2g}}, \tag{I2}$$

where

$$\kappa_g = \frac{23 \times 3^{2g} + 1}{8}3^f. \tag{I3}$$

For f being even, the Pauli strings $\mathcal{L}_j^{(f)}$ and $\mathcal{R}_k^{(f)}$ can be explicitly expressed as

$$\mathcal{L}_j^{(f)} = \prod_{g=0}^{(d-5-f)/2} X_{j-\lambda_g-9\times 3^{f+2g}}X_{j-\lambda_g}, \tag{I4}$$

$$\mathcal{R}_k^{(f)} = \prod_{g=0}^{(d-5-f)/2} X_{k+\lambda_g}X_{k+\lambda_g+9\times 3^{f+2g}}, \tag{I5}$$

where

$$\lambda_g = \frac{69 \times 3^{2g} - 13}{8}3^f. \tag{I6}$$

The Pauli strings $\mathcal{L}_j^{(f)}$ and $\mathcal{R}_k^{(f)}$ involve $2\lfloor (d-f-2)/2 \rfloor$ Pauli X_l operators at layer f.

Backpropagation of products (G7) of Pauli strings. We now show that the product $H_{jk}^{(f)}$ appears at every layer $f < d - 2$, see Table III in Appendix G. We start with $H_{(\frac{N+1}{2}-3^{d-2})(\frac{N+1}{2}+3^{d-2})}^{(d-2)}$ at layer $f = d - 2$. The product $H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$ backpropagates through the X-error correcting layer $\widetilde{\mathrm{QEC}}_f^X$ according to the recursion relation

$$
\begin{aligned}
&\widetilde{\mathrm{QEC}}_f^{X\,\dagger} H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)} \widetilde{\mathrm{QEC}}_f^X \\
&\rightarrow H_{(\frac{N+1}{2}-l_{f-1}\times 3^{f-1})(\frac{N+1}{2}+l_{f-1}\times 3^{f-1})}^{(f-1)},
\end{aligned} \tag{I7}
$$

where $l_{f-1} = 3l_f + 2$. Due to the distance $|l - m| \geqslant 3 \times 3^f$ between every pair of Pauli operators X_l and X_m in the strings $\mathcal{L}_{\frac{N+1}{2}-l_f\times 3^f}^{(f)}$ and $\mathcal{R}_{\frac{N+1}{2}+l_f\times 3^f}^{(f)}$, each Pauli X_l operator separately backpropagates through the odd X-error correcting layer $\widetilde{\mathrm{QEC}}_f^X$ according to Eq. (G3). The Pauli strings $G_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$ and $G_{[\frac{N+1}{2}-(l_f-1)\times 3^f][\frac{N+1}{2}+(l_f-1)\times 3^f]}^{(f)}$ in the product $H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$ also separately backpropagate through the X-error correcting layer $\widetilde{\mathrm{QEC}}_f^X$ according to Eq. (G3). As a result, backpropagating the product $H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$ through the X-error correcting layer $\widetilde{\mathrm{QEC}}_f^X$ gives rise to a sum of $16^{2(d-f-1)}$ terms including the product $H_{(\frac{N+1}{2}-l_{f-1}\times 3^{f-1})(\frac{N+1}{2}+l_{f-1}\times 3^{f-1})}^{(f-1)}$, where we used Eqs. (G8) and (G9). We focus on the single product $H_{(\frac{N+1}{2}-l_{f-1}\times 3^{f-1})(\frac{N+1}{2}+l_{f-1}\times 3^{f-1})}^{(f-1)}$ as indicated by the recursion relation (I7) and displayed in Table III.

The product $H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$ backpropagates through the Z-error correcting layer $\widetilde{\mathrm{QEC}}_f^Z$ according to the recursion relation

$$
\begin{aligned}
&\widetilde{\mathrm{QEC}}_f^{Z\,\dagger} H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)} \widetilde{\mathrm{QEC}}_f^Z \\
&\rightarrow H_{(\frac{N+1}{2}-l_{f-1}\times 3^{f-1})(\frac{N+1}{2}+l_{f-1}\times 3^{f-1})}^{(f-1)},
\end{aligned} \tag{I8}
$$

where $l_{f-1} = 3l_f - 5$. Each Pauli string in the product $H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$ backpropagates separately through the Z-error correcting layer $\widetilde{\mathrm{QEC}}_f^Z$ according to Eq. (G4) giving rise to a sum of $4^{2(l_f+d-f)-5}$ terms including the product $H_{\frac{N+1}{2}-l_{f-1}\times 3^{f-1})(\frac{N+1}{2}+l_{f-1}\times 3^{f-1})}^{(f-1)}$ as indicated by the recursion relation (I8) and stated in Table III. To show this, we distribute the parentheses in Eq. (G4) and exploit Eqs. (G10) and (G11) as well as that

$$
\begin{aligned}
G_{jk}^{(f-1)} = {}&X_{j+2\times 3^{f-1}}X_{j+4\times 3^{f-1}}X_{j+8\times 3^{f-1}}\left(\prod_{\delta'} X_{\delta'-\epsilon}X_{\delta'+\epsilon}\right) \\
&\times \left(\prod_{\delta''} X_{\delta''}\right) X_{k-8\times 3^{f-1}}X_{k-4\times 3^{f-1}}X_{k-2\times 3^{f-1}},
\end{aligned} \tag{I9}
$$

and

$$
\begin{aligned}
G_{(j+3^{f-1})(k-3^{f-1})}^{(f-1)} = {}&X_{j+3^{f-1}}X_{j+5\times 3^{f-1}}X_{j+11\times 3^{f-1}} \\
&\times \left(\prod_{\delta'} X_{\delta'}\right)\left(\prod_{\delta''} X_{\delta''-\epsilon}X_{\delta''+\epsilon}\right) \\
&\times X_{k-11\times 3^{f-1}}X_{k-5\times 3^{f-1}}X_{k-3^{f-1}},
\end{aligned} \tag{I10}
$$

where $\delta' \in \{j + 7 \times 3^{f-1}, j + 13 \times 3^{f-1}, ..., k - 7 \times 3^{f-1}\}$, $\delta'' \in \{j + 10 \times 3^{f-1}, j + 16 \times 3^{f-1}..., k - 10 \times 3^{f-1}\}$, and $\epsilon = 7 \times 3^{f-1}$.

To summarize the backpropagation of the products $H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$ in the QCNN circuit, we start with the first product $H_{(\frac{N+1}{2}-3^{d-2})(\frac{N+1}{2}+3^{d-2})}^{(d-2)}$ at layer $f = d - 2$, cf. Table III. Using the recursion relations (I7) and (I8) for odd layers $\widetilde{\mathrm{QEC}}_f^X$ and even layers $\widetilde{\mathrm{QEC}}_f^Z$, respectively, we obtain the product $H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$ at every layer $f < d - 2$. Note that the product $H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$ at layer f emerges only by backpropagating the product $H_{(\frac{N+1}{2}-l_{f+1}\times 3^{f+1})(\frac{N+1}{2}+l_{f+1}\times 3^{f+1})}^{(f+1)}$ from layer $f + 1$. The recursion relations (G3) and (G4) dictate that all other products of Pauli strings at layer $f + 1$ give rise to products at layer f different to $H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$.

We now investigate the length l_f of the Pauli string $G_{[\frac{N+1}{2}-(l_f-1)\times 3^f][\frac{N+1}{2}+(l_f-1)\times 3^f]}^{(f)}$ in the product $H_{(\frac{N+1}{2}-l_f\times 3^f)(\frac{N+1}{2}+l_f\times 3^f)}^{(f)}$ at every layer f. We start with $l_{d-2} = 1$ at layer $f = d - 2$, see Table III. By backpropagating through the X-error correcting layer, the length $l_{f-1} = 3l_f + 2$ of the Pauli string $G_{[\frac{N+1}{2}-(l_{f-1}-1)\times 3^{f-1}][\frac{N+1}{2}+(l_{f-1}-1)\times 3^{f-1}]}^{(f-1)}$ at layer $f - 1$ increases by a factor of three compared to the length l_f of the Pauli string $G_{[\frac{N+1}{2}-(l_f-1)\times 3^f][\frac{N+1}{2}+(l_f-1)\times 3^f]}^{(f)}$ at layer f, see Eq. (I7). By backpropagating through the Z-error correcting layer, the length $l_{f-1} = 3l_f - 5$ also increases by a factor of three, see Eq. (I8). After two successive layers—the first layer correcting X errors and the second layer correcting Z errors—the length of the Pauli string increases by a factor of nine from l_f to $l_{f-2} = 9l_f + 1$. At every odd layer f, the length can be expressed as a sum

$$
l_f = \sum_{g=0}^{(d-f-2)/2} 9^g = \frac{3^{d-f} - 1}{8}. \tag{I11}
$$

At every even layer f, the length can be expressed as $l_f = \frac{3^{d-f}+13}{8}$, cf. Table III.

The product $H_{(\frac{N+1}{2}-l_0)(\frac{N+1}{2}+l_0)}^{(0)}$ at layer $f = 0$ is mapped by the disentangling unitary $U_N^\dagger$ onto the product of SOPs

$$
U_N H_{(\frac{N+1}{2}-l_0)(\frac{N+1}{2}+l_0)}^{(0)} U_N^\dagger = \tilde{\mathcal{L}} S_{(i-l_0-1)(i+l_0+1)} S_{(i-l_0)(i+l_0)} \tilde{\mathcal{R}}, \tag{I12}
$$

see Table III where

$$
\tilde{\mathcal{L}} = U_N \mathcal{L}^{(0)}_{\frac{N+1}{2}-l_0} U_N^\dagger
$$

$$
= \prod_{g=0}^{(d-5)/2} C_{\frac{N+1}{2}-l_0-\lambda_g-9\times 3^{2g}} C_{\frac{N+1}{2}-l_0-\lambda_g}, \quad \text{(I13)}
$$

$$
\tilde{\mathcal{R}} = U_N \mathcal{R}^{(0)}_{\frac{N+1}{2}+l_0} U_N^\dagger
$$

$$
= \prod_{g=0}^{(d-5)/2} C_{\frac{N+1}{2}+l_0+\lambda_g} C_{\frac{N+1}{2}+l_0+\lambda_g+9\times 3^{2g}}. \quad \text{(I14)}
$$

Derivation of Eq. (G13). In Eq. (G13), we backpropagate the product $H^{(2)}_{(\frac{N+1}{2}-9l_2)(\frac{N+1}{2}+9l_2)}$ that appears at layer $f=1$ through the first X-error correcting layer $\widetilde{\text{QEC}}_1^X$ and the disentangling unitary $U_N^\dagger$ using the recursion relation (G3) and Eq. (G6), respectively. Note that every Pauli operator X_ζ in the product $H^{(2)}_{(\frac{N+1}{2}-9l_2)(\frac{N+1}{2}+9l_2)}$ separately backpropagates through the X-error correcting layer $\widetilde{\text{QEC}}_1^X$ due to the distance at least 9 between these Pauli operators. The Pauli strings $G^{(2)}_{[\frac{N+1}{2}-9(l_2-1)][\frac{N+1}{2}+9(l_2-1)]}$ and $G^{(2)}_{(\frac{N+1}{2}-9l_2)(\frac{N+1}{2}+9l_2)}$ in the product $H^{(2)}_{jk}$ involve $l_2 = \frac{3^{d-2}+13}{8}$ and l_2+1, respectively, Pauli X_ζ operators. Backpropagating these Pauli strings through the layer $\widetilde{\text{QEC}}_1^X$ and the disentangling unitary $U_N^\dagger$ gives rise to the product

$$
U_N \widetilde{\text{QEC}}_1^{X\,\dagger} G^{(2)}_{(\frac{N+1}{2}-9l_2)(\frac{N+1}{2}+9l_2)}
$$

$$
\times\, G^{(2)}_{[\frac{N+1}{2}-9(l_2-1)][\frac{N+1}{2}+9(l_2-1)]} \widetilde{\text{QEC}}_1^X U_N^\dagger = \prod_\zeta E_\zeta \quad \text{(I15)}
$$

of $2l_2 + 1 = \frac{3^{d-2}+17}{4}$ terms $E_\zeta = A_\zeta C_\zeta B_\zeta$ where $\zeta \in \{\frac{N+1}{2} - 9l_2, \frac{N+1}{2} - 9(l_2-1), ..., \frac{N+1}{2} + 9l_2\}$ and $C_\zeta = S_{(\zeta-1)(\zeta+1)}$. Operators A_ζ and B_ζ are defined in Eqs. (G14) and (G15). Backpropagating the Pauli strings $\mathcal{L}^{(2)}_{\frac{N+1}{2}-9l_2}$ and $\mathcal{R}^{(2)}_{\frac{N+1}{2}+9l_2}$ through the layer $\widetilde{\text{QEC}}_1^X$ and the disentangling unitary $U_N^\dagger$ gives rise to the products

$$
\bar{\mathcal{L}} = U_N \widetilde{\text{QEC}}_1^{X\,\dagger} \mathcal{L}^{(2)}_{\frac{N+1}{2}-9l_2} \widetilde{\text{QEC}}_1^X U_N^\dagger
$$

$$
= \prod_{g=0}^{(d-7)/2} E_{\frac{N+1}{2}-9l_2-\lambda_g-81\times 3^{2g}} E_{\frac{N+1}{2}-9l_2-\lambda_g}, \quad \text{(I16)}
$$

$$
\bar{\mathcal{R}} = U_N \widetilde{\text{QEC}}_1^{X\,\dagger} \mathcal{R}^{(2)}_{\frac{N+1}{2}+9l_2} \widetilde{\text{QEC}}_1^X U_N^\dagger
$$

$$
= \prod_{g=0}^{(d-7)/2} E_{\frac{N+1}{2}+9l_2+\lambda_g} E_{\frac{N+1}{2}+9l_2+\lambda_g+81\times 3^{2g}}. \quad \text{(I17)}
$$

By combining Eqs. (I15)–(I17) we obtain Eq. (G13).

[1] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. S. L. Brandao, D. A. Buell *et al.*, Quantum supremacy using a programmable superconducting processor, Nature (London) **574**, 505 (2019).

[2] R. P. Feynman, Simulating physics with computers, Int. J. Theor. Phys. **21**, 467 (1982).

[3] Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. D. Sawaya *et al.*, Quantum chemistry in the age of quantum computing, Chem. Rev. **119**, 10856 (2019).

[4] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, Quantum machine learning, Nature (London) **549**, 195 (2017).

[5] H.-Y. Huang, R. Kueng, and J. Preskill, Predicting many properties of a quantum system from very few measurements, Nat. Phys. **16**, 1050 (2020).

[6] S. Lloyd, M. Mohseni, and P. Rebentrost, Quantum principal component analysis, Nat. Phys. **10**, 631 (2014).

[7] J. Romero, J. P. Olson, and A. Aspuru-Guzik, Quantum autoencoders for efficient compression of quantum data, Quantum Sci. Technol. **2**, 045001 (2017).

[8] D. Bondarenko and P. Feldmann, Quantum autoencoders to denoise quantum data, Phys. Rev. Lett. **124**, 130502 (2020).

[9] X.-M. Zhang, W. Kong, M. U. Farooq, M.-H. Yung, G. Guo, and X. Wang, Generic detection-based error mitigation using quantum autoencoders, Phys. Rev. A **103**, L040403 (2021).

[10] N. Wiebe, C. Granade, C. Ferrie, and D. G. Cory, Hamiltonian learning and certification using quantum resources, Phys. Rev. Lett. **112**, 190501 (2014).

[11] A. A. Gentile, B. Flynn, S. Knauer, N. Wiebe, S. Paesani, C. E. Granade, J. G. Rarity, R. Santagati, and A. Laing, Learning models of quantum systems from experiments, Nat. Phys. **17**, 837 (2021).

[12] S. Ghosh, A. Opala, M. Matuszewski, T. Paterek, and T. C. H. Liew, Quantum reservoir processing, npj Quantum Inf. **5**, 35 (2019).

[13] E. Farhi and H. Neven, Classification with quantum neural networks on near term processors, arXiv:1802.06002.

[14] I. Cong, S. Choi, and M. D. Lukin, Quantum convolutional neural networks, Nat. Phys. **15**, 1273 (2019).

[15] K. Beer, D. Bondarenko, T. Farrelly, T. J. Osborne, R. Salzmann, D. Scheiermann, and R. Wolf, Training deep quantum neural networks, Nat. Commun. **11**, 808 (2020).

[16] K. Kottmann, F. Metz, J. Fraxanet, and N. Baldelli, Variational quantum anomaly detection: Unsupervised mapping of phase diagrams on a physical quantum computer, Phys. Rev. Res. **3**, 043184 (2021).

[17] M. Gong, H.-L. Huang, S. Wang, C. Guo, S. Li, Y. Wu, Q. Zhu, Y. Zhao, S. Guo, H. Qian *et al.*, Quantum neuronal sensing of quantum many-body states on a 61-qubit programmable superconducting processor, Sci. Bull. **68**, 906 (2023).

[18] F. Pollmann, A. M. Turner, E. Berg, and M. Oshikawa, Entanglement spectrum of a topological phase in one dimension, Phys. Rev. B **81**, 064439 (2010).

[19] X. Chen, Z.-C. Gu, and X.-G. Wen, Classification of gapped symmetric phases in one-dimensional spin systems, Phys. Rev. B **83**, 035107 (2011).

[20] S. de Léséleuc, V. Lienhard, P. Scholl, D. Barredo, S. Weber, N. Lang, H. P. Büchler, T. Lahaye, and A. Browaeys,

Observation of a symmetry-protected topological phase of interacting bosons with Rydberg atoms, Science **365**, 775 (2019).

[21] G. Semeghini, H. Levine, A. Keesling, S. Ebadi, T. T. Wang, D. Bluvstein, R. Verresen, H. Pichler, M. Kalinowski, R. Samajdar *et al.*, Probing topological spin liquids on a programmable quantum simulator, Science **374**, 1242 (2021).

[22] S. Sachdev, *Quantum Phase Transitions* (Cambridge University Press, Cambridge, 2011).

[23] Z. F. Wang, H. Zhang, D. Liu, C. Liu, C. Tang, C. Song, Y. Zhong, J. Peng, F. Li, C. Nie *et al.*, Topological edge states in a high-temperature superconductor FeSe/SrTiO$_3$(001) film, Nat. Mater. **15**, 968 (2016).

[24] J. Carrasquilla and R. G. Melko, Machine learning phases of matter, Nat. Phys. **13**, 431 (2017).

[25] E. P. L. van Nieuwenburg, Y.-H. Liu, and S. D. Huber, Learning phase transitions by confusion, Nat. Phys. **13**, 435 (2017).

[26] E. Greplova, A. Valenti, G. Boschung, F. Schäfer, N. Lörch, and S. D. Huber, Unsupervised identification of topological phase transitions using predictive models, New J. Phys. **22**, 045003 (2020).

[27] B. S. Rem, N. Käming, M. Tarnowski, L. Asteria, N. Fläschner, C. Becker, K. Sengstock, and C. Weitenberg, Identifying quantum phase transitions using artificial neural networks on experimental data, Nat. Phys. **15**, 917 (2019).

[28] A. Bohrdt, S. Kim, A. Lukin, M. Rispoli, R. Schittko, M. Knap, M. Greiner, and J. Léonard, Analyzing nonequilibrium quantum states through snapshots with artificial neural networks, Phys. Rev. Lett. **127**, 150504 (2021).

[29] N. Käming, A. Dawid, K. Kottmann, M. Lewenstein, K. Sengstock, A. Dauphin, and C. Weitenberg, Unsupervised machine learning of topological phase transitions from experimental data, Mach. Learn.: Sci. Technol. **2**, 035037 (2021).

[30] C. Miles, R. Samajdar, S. Ebadi, T. T. Wang, H. Pichler, S. Sachdev, M. D. Lukin, M. Greiner, K. Q. Weinberger, and E.-A. Kim, Machine learning discovery of new phases in programmable quantum simulator snapshots, Phys. Rev. Res. **5**, 013026 (2023).

[31] A. Smith, B. Jobst, A. G. Green, and F. Pollmann, Crossing a topological phase transition with a quantum computer, Phys. Rev. Res. **4**, L022020 (2022).

[32] K. J. Satzinger *et al.*, Realizing topologically ordered states on a quantum processor, Science **374**, 1237 (2021).

[33] M. Iqbal, N. Tantivasadakarn, T. M. Gatterman, J. A. Gerber, K. Gilmore, D. Gresh, A. Hankin, N. Hewitt, C. V. Horst, M. Matheny *et al.*, Topological order from measurements and feedforward on a trapped ion quantum computer, Commun. Phys. **7**, 205 (2024).

[34] D. Azses, R. Haenel, Y. Naveh, R. Raussendorf, E. Sela, and E. G. Dalla Torre, Identification of symmetry-protected topological states on noisy quantum computers, Phys. Rev. Lett. **125**, 120502 (2020).

[35] D. Pérez-García, M. M. Wolf, M. Sanz, F. Verstraete, and J. I. Cirac, String order and symmetries in quantum spin lattices, Phys. Rev. Lett. **100**, 167202 (2008).

[36] F. Pollmann and A. M. Turner, Detection of symmetry-protected topological phases in one dimension, Phys. Rev. B **86**, 125441 (2012).

[37] I. Cong, N. Maskara, M. C. Tran, H. Pichler, G. Semeghini, S. F. Yelin, S. Choi, and M. D. Lukin, Enhancing detection of topological order by local error correction, Nat. Commun. **15**, 1527 (2024).

[38] H.-Y. Huang, R. Kueng, G. Torlai, V. V. Albert, and J. Preskill, Provably efficient machine learning for quantum many-body problems, Science **377**, eabk3333 (2022).

[39] M. C. Caro, H.-Y. Huang, M. Cerezo, K. Sharma, A. Sornborger, L. Cincio, and P. J. Coles, Generalization in quantum machine learning from few training data, Nat. Commun. **13**, 4919 (2022).

[40] Y.-J. Liu, A. Smith, M. Knap, and F. Pollmann, Model-independent learning of quantum phases of matter with quantum convolutional neural networks, Phys. Rev. Lett. **130**, 220603 (2023).

[41] E. Lake, S. Balasubramanian, and S. Choi, Exact quantum algorithms for quantum phase recognition: Renormalization group and error correction, arXiv:2211.09803.

[42] J. Herrmann, S. M. Llima, A. Remm, P. Zapletal, N. A. McMahon, C. Scarato, F. Swiadek, C. K. Andersen, C. Hellings, S. Krinner *et al.*, Realizing quantum convolutional neural networks on a superconducting quantum processor to recognize quantum phases, Nat. Commun. **13**, 4144 (2022).

[43] C. de Groot, A. Turzillo, and N. Schuch, Symmetry protected topological order in open quantum systems, Quantum **6**, 856 (2022).

[44] R. Verresen, R. Moessner, and F. Pollmann, One-dimensional symmetry protected topological phases and their transitions, Phys. Rev. B **96**, 165124 (2017).

[45] J. Hauschild and F. Pollmann, Efficient numerical simulations with tensor networks: Tensor network Python (TeNPy), SciPost Phys. Lect. Notes 5 (2018).

[46] We will show in Sec. VII that the error tolerance is limited close to phase boundaries as the threshold error probability decreases with diverging correlation lengths.

[47] J. Eisert, M. Cramer, and M. B. Plenio, *Colloquium* : Area laws for the entanglement entropy, Rev. Mod. Phys. **82**, 277 (2010).

[48] The correlation length corresponds to a characteristic length scale at which quantum correlation functions exponentially decay. For matrix product states with the unique largest eigenvalue $|\eta_1| = 1$ of the corresponding transfer matrix, the correlation length $\xi \propto -\frac{1}{\log |\eta_2|}$ is determined by the second largest eigenvalue η_2.

[49] A. Yu. Kitaev, Fault-tolerant quantum computation by anyons, Ann. Phys. **303**, 2 (2003).

[50] A. Mesaros and Y. Ran, Classification of symmetry enriched topological phases with exactly solvable models, Phys. Rev. B **87**, 155115 (2013).

[51] Y. Ge, A. Molnár, and J. I. Cirac, Rapid adiabatic preparation of injective projected entangled pair states and Gibbs states, Phys. Rev. Lett. **116**, 080503 (2016).

[52] Y.-J. Liu, K. Shtengel, A. Smith, and F. Pollmann, Methods for simulating string-net states and anyons on a digital quantum computer, PRX Quantum **3**, 040315 (2022).

[53] G. Duclos-Cianci and D. Poulin, Fast decoders for topological quantum codes, Phys. Rev. Lett. **104**, 050504 (2010).

[54] A. Feiguin, S. Trebst, A. W. W. Ludwig, M. Troyer, A. Kitaev, Z. Wang, and M. H. Freedman, Interacting anyons in topological quantum liquids: The golden chain, Phys. Rev. Lett. **98**, 160409 (2007).

[55] L. Savary and L. Balents, Quantum spin liquids: A review, Rep. Prog. Phys. **80**, 016502 (2017).

[56] A. Pesah, M. Cerezo, S. Wang, T. Volkoff, A. T. Sornborger, and P. J. Coles, Absence of barren plateaus in quantum convolutional neural networks, Phys. Rev. X **11**, 041011 (2021).

[57] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, A variational eigenvalue solver on a photonic quantum processor, Nat. Commun. **5**, 4213 (2014).

[58] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, The theory of variational hybrid quantum-classical algorithms, New J. Phys. **18**, 023023 (2016).