

Deep learning Local Reduced Density Matrices for Many-body Hamiltonian Estimation

Xinran Ma,¹ Z. C. Tu,¹ and Shi-Ju Ran^{2,*}

¹*Department of Physics, Beijing Normal University, Beijing 100875, China*

²*Department of Physics, Capital Normal University, Beijing 100048, China*

(Dated: December 19, 2021)

Human experts cannot efficiently access the physical information of quantum many-body states by simply “reading” the coefficients, but have to rely on the previous knowledge such as order parameters and quantum measurements. In this work, we demonstrate that convolutional neural network (CNN) can learn from the coefficients of local reduced density matrices to estimate the physical parameters of the many-body Hamiltonians, such as coupling strengths and magnetic fields, provided the states as the ground states. We propose QubismNet that consists of two main parts: the Qubism map that visualizes the ground states (or the purified reduced density matrices) as images, and a CNN that maps the images to the target physical parameters. By assuming certain constraints on the training set for the sake of balance, QubismNet exhibits impressive powers of learning and generalization on several quantum spin models. While the training samples are restricted to the states from certain ranges of the parameters, QubismNet can accurately estimate the parameters of the states beyond such training regions. For instance, our results show that QubismNet can estimate the magnetic fields near the critical point by learning from the states away from the critical vicinity. Our work illuminates a data-driven way to infer the Hamiltonians that give the designed ground states, and therefore would benefit the existing and future generalizations of quantum technologies such as Hamiltonian-based quantum simulations and state tomography.

INTRODUCTION

Machine learning (ML) has recently been applied to various issues that are difficult using purely the “conventional” techniques in physics (for instance, tensor network [1–3], quantum Monte Carlo [4, 5], and etc.). The successful applications include identifying the classical/quantum phases and topologies without computing order parameters [6–12], predicting physical properties of materials [13–15], efficiently representing non-trivial quantum states [16–19], to name but a few.

Among others, obtaining the eigenstates, particularly the ground states, of a given quantum many-body Hamiltonian belongs to the central topics in the contemporary physics [20, 21]. The inverse problems, which are of equal significance and practicality, are much less studied due to the lack of valid methods. ML serves as a novel approach that has recently gained certain inspiring successes in such problems [22–25]. In particular, one important issue under hot debate is to access the information of the potentials or interactions by learning from physical data. For instance, Xin *et al* utilized fully-connected neural network to recover the ground states of k -local Hamiltonians from local measurements [26]. Hegde *et al* employed the kernel ridge regression to achieve accurate and transferable predictions of Hamiltonians for a variety of material environments [27]. Li *et al* identified the effective Hamiltonians in magnetic systems and extracted the dominant spin interactions in MnO and TbMnO₃ through multiple linear regression [28]. Sehanobish *et al* proposed the quantum potential neural networks to reconstruct the effective potential given the wave functions [29].

However, most existing works in this direction utilize the regression methods or shallow neural networks, which usually possess relative low learning or generalizing powers. In ML, one usually uses deep networks, such as convolutional neural network (CNN) [30, 31], to solve sophisticated problems such

as the classifications of real-life images. The excellent learning and generalization abilities of CNN have been widely recognized in numerous applications in computer sciences (c.f. Refs. [32–34] for instance). Recently, Berthusen *et al* utilized CNN to extract the crystal field Stevens parameters from the thermodynamic data, which illustrates the validity of CNN-based method in deducing physical information [35]. Goh *et al* put forward a deep CNN model named Chemception to predict chemical properties with the 2D drawings of molecules [36]. Laanait *et al* used an encoder-decode architecture with convolutional layers to generate the local electron density of material by learning from the diffraction patterns [37]. It is interesting and unexplored whether CNN is capable of solving more challenging issues, including those with the presence of strong correlations and many-body effects.

In this work, the problem we consider is inverse to those of solving the eigenstates of a given Hamiltonian. Suppose the state or its local reduced density matrices (RDMs) is given. Our aim is to estimate the parameters in the many-body Hamiltonian by training a CNN model, so that the given state is the ground state. Solving such a problem would be meaningful and important for, e.g., designing the Hamiltonian of a quantum annealer to prepare a target state [38]. To this end, we propose QubismNet that consists of two main parts (FIG. 1 (a)). The first part is a map to transform the RDMs into images, and the second is a CNN to transform the images to the estimations of the target parameters in the Hamiltonian. The purpose of mapping the RDMs to images is to utilize the power of CNN processing images. Similar idea has been used in Ref. [35], where the thermodynamic data (specific heat and others) are transformed to images by wavelet transformation before being fed to the CNN. The RDM-image map we use is known as Qubism [39], where the obtained images are of fractals (see some examples in FIG. 1 (b)) that can reveal the physical properties of the state.

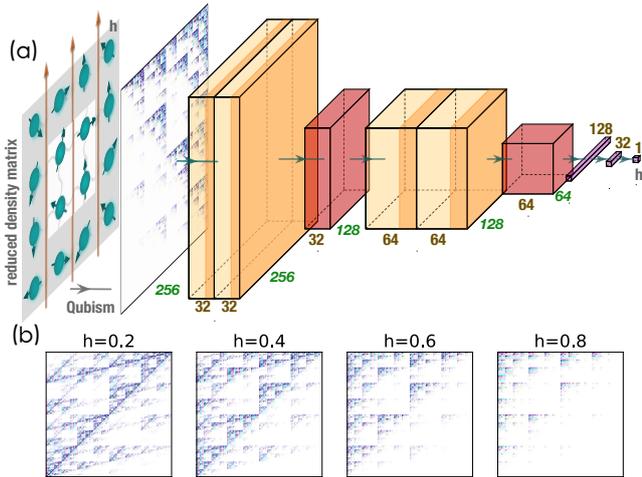


FIG. 1. (Color online) (a) Illustration of QubismNet. Its first part is the Qubism map that transforms a quantum wave-function to an image of fractals. The second part is a convolutional neural network that maps the images to the estimations of the target parameters. (b) Examples of the images obtained by Qubism. The ground states are from the XY model with the magnetic fields $h = 0.2 \sim 0.8$.

We benchmark QubismNet on several quantum spin models defined on 1D and 2D lattices. The learning and generalization powers of QubismNet are tested by dividing the samples (i.e., the ground states taking different values of certain parameter in the Hamiltonian) into testing and generalizing sets. The parameters corresponding to the states in the testing set are independently and identically distributed (*i.i.d.*) as the states in the training set. QubismNet can estimate the parameters of such states with high performance. The parameters corresponding to the states in the generalizing set are restricted in a certain range in which no training states are taken. To keep the training data balanced, the training samples are taken from the boundaries of the whole parameter space, and the generalizing samples are taken from a sub-region in the middle of the parameter space. Our results show that QubismNet can generalize what it has learned from the training set to estimate the parameters of the generalizing states with fair performance. For instance, QubismNet only learns from the states away from the critical point and well estimates the magnetic fields given the RDMs of the states in the critical vicinity. Our work suggests that CNN is capable of extracting information directly from the coefficients of the RDMs, while human experts have to rely on the previous knowledge such as order parameters and measurements.

QUBISMNET: ESTIMATING PHYSICAL PARAMETERS FROM LOCAL REDUCED DENSITY MATRICES

QubismNet consists of two main parts. The first part is a Qubism map [39] that was originally proposed to transform states to images of fractals in a one-to-one way.

Taking the quantum Ising model (QIM) in a transverse magnetic field as an example, the Hamiltonian reads

$$\hat{H}(h) = J \sum_{\langle i,j \rangle} \hat{S}_i^z \hat{S}_j^z - h \sum_{k=1}^L \hat{S}_k^x, \quad (1)$$

with \hat{S}^α the α -component spin operator ($\alpha = x, z$) and L the system size. Here, we take the coupling constant $J = 1$ as the energy scale. There are several well-established methods to calculate the ground states given the Hamiltonians, such as density matrix renormalization group (DMRG) [40, 41], tensor network algorithms [1–3], quantum Monte Carlo [4, 5], and etc. This work considers an inverse problem: estimating the magnetic fields h (or other parameters) given the RDMs of the ground states.

In specific, we denote the training set as $\{|\psi_m\rangle\}$ ($m = 1, \dots, N_{\text{train}}$), where $|\psi_m\rangle$ is the ground state of $\hat{H}(h_m)$. To train the CNN whose output is the value of the target parameter, we choose the mean-square error (MSE) as the loss function

$$\varepsilon = \frac{1}{N_{\text{train}}} \sum_{m=1}^{N_{\text{train}}} (h_m^p - h_m)^2, \quad (2)$$

with h_m^p the estimation of the magnetic field of $|\psi_m\rangle$ by the QubismNet. The variational parameters in the CNN are optimized by minimizing the loss function using the gradient method. We choose RMSProp [42] as the optimizer to control the gradient steps. More details of the Qubism map and CNN are provided in the Supplementary Material.

To benchmark the generalization power of QubismNet, we introduce the testing and generalizing sets [43]. The testing sets contain the states whose magnetic fields are different from but *i.i.d.* with the training states. The states in the generalizing set are different from both the training and testing states, and are distributed in a different region. For instance, we uniformly choose N_{train} values of h within $0 < h < 0.5 - \delta/2$ and $0.5 + \delta/2 < h < 1$ for the training set, and choose other N_{test} values of h in the same regions of h for the testing set. For the generalizing set, we uniformly choose N_g values of h within $0.5 - \delta/2 < h < 0.5 + \delta/2$. We dub δ as *generalization width*. Note $h = 0.5$ is the quantum critical point of QIM. The states in either the testing or the generalizing sets are not used to train the CNN.

For the large-size systems, it is inefficient to directly apply the Qubism map, as it requires the full coefficients of the quantum states. To avoid such a problem, we bring in the RDMs combined with purification. In specific, we choose a subsystem of a moderate size (denoted by L_b) and calculate the reduced density matrix $\hat{\rho}(|\psi_m\rangle) = \text{Tr}_s |\psi_m\rangle\langle\psi_m|$ with Tr_s tracing the degrees of freedom in the subsystem. In general, if L_b is comparable or larger than the correlation length, the RDM would contain the dominant physical information of the whole system [44, 45]. Interestingly, according to our simulations, it is even not necessary to set L_b larger than the correlation length to accurately estimate the physical parameters from the RDMs.

To map a RDM to image by Qubism, we write it as a pure state as $\hat{\rho}(|\psi_m\rangle) = \sum_{ii'} \rho_{ii'} |i\rangle\langle i'| \rightarrow |\rho_m\rangle = \sum_{ii'} \rho_{ii'} |ii'\rangle$. One can see that $|\rho_m\rangle$ is the purification of $\hat{\rho}(|\psi_m\rangle)^2$ since we have $\text{Tr}_{i'} |\rho_m\rangle\langle\rho_m| = \hat{\rho}(|\psi_m\rangle)^2$. Therefore, we feed the QubismNet by $\{|\rho_m\rangle\}$, which contain identical amount of information as $\{\hat{\rho}(|\psi_m\rangle)\}$. The parameter complexity of $|\rho_m\rangle$ is independent on the size of the whole system. It is same as the complexity of a state with $2L_b$ spins.

RESULTS AND DISCUSSIONS

We first benchmark the training and testing accuracies of QubismNet on one-dimensional (1D) QIM by taking $L = 64$ as the system size and $L_b = 8$ as the subsystem size in the RDM trick. We take the periodic boundary condition, meaning the first and last spins are interacted as nearest neighbors. FIG. 2 (a) shows the estimated fields h^p against the true fields h . The $N_{\text{train}} = 1000$ training states are taken as the ground states by uniformly choosing different magnetic fields in $0 < h < 1$. The h 's of the $N_{\text{test}} = 100$ testing states are also uniformly taken in $0 < h < 1$, which are different from the fields of the training states. No generalizing states are taken (i.e., $\delta = 0$). The QubismNet accurately estimates the magnetic fields of both the training and testing states. We have the testing error $\epsilon_t \simeq 1.21 \times 10^{-4}$ evaluated by the loss function, i.e., MSE, of all testing states. The estimations are accurate near the critical point, where the states possess relatively long-range correlations.

We also test the QubismNet on 1D XXZ model with periodic boundary condition. We consider two cases, whose Hamiltonians are written, respectively, as

$$\hat{H}(J_z) = \sum_{\langle i,j \rangle} (\hat{S}_i^x \hat{S}_j^x + \hat{S}_i^y \hat{S}_j^y + J_z \hat{S}_i^z \hat{S}_j^z), \quad (3)$$

$$\hat{H}(h) = \sum_{\langle i,j \rangle} (\hat{S}_i^x \hat{S}_j^x + \hat{S}_i^y \hat{S}_j^y) + h \sum_{k=1}^L \hat{S}_k^z. \quad (4)$$

The physical parameters to be estimated by the QubismNet are the coupling strength J_z in Eq. (3) and the longitudinal field h in Eq. (4). We dub the latter as XY model, where we take zero J_z and non-zero longitudinal field. We use the RDM trick with $L = 64$ and $L_b = 8$. The testing accuracies of the XXZ and XY models are about $\epsilon_t \sim O(10^{-5})$.

To benchmark the generalization power, we set $\delta = 0.4$ [FIG. 2 (b)]. Within $0.3 < h < 0.7$ (the light yellow shadow), no training states are taken. In this range, we averagely take $N_g = 40$ h 's with an interval $dh = 0.01$ as the generalizing set. For the QIM, a quantum phase transition occurs at $h = 0.5$. In our setting, the QubismNet only learns from the states away from the critical vicinity. Our results show that it can generalize from what it has learned and estimate the magnetic fields near the critical point. We have the generalizing error (the MSE evaluated by the generalizing set) $\epsilon_g \sim O(10^{-3})$ using the RDM trick with $L = 64$ and $L_b = 8$.

For the XY and XXZ models, the system is in the gapless phase for $0 < h < 1$ [46]. We set the same ranges for the training, testing and generalizing sets as above. Without the RDM trick, we take $L = 16$, and find that ‘‘stages’’ appear in the h - h^p curves. This is due to the energy gaps caused by the finite-size effects. For instance, by changing h of the XY model from 0.15 to 0.28, no energy level crossing occurs. It means the ground states within this range is the same state. Such ‘‘stages’’ obviously hinder the estimation of the physical parameters from the ground states since QubismNet distinguishes no differences from the states by varying h within a stage. We propose to resolve this issue by increasing L (e.g., to $L = 64$). The RDM trick has to be used since we cannot handle the full 2^{64} coefficients in the Qubism map. From our results, the ‘‘stages’’ of the h - h^p curves are largely suppressed using the RDM trick. The testing and generalizing errors are decreased by more than 100 and 20 times, respectively. QubismNet is also tested on the frustrated breathing kagome antiferromagnet [47, 48], where both the testing and generalizing errors are around $O(10^{-4})$. See more details in the Supplementary Material.

We also test on the XXZ and XY models on (4×16) 2D square lattice with periodic boundary condition. The subsystem for the RDM is chosen in the middle of lattice with the size 4×2 . The 2D XY model, whose local Hamiltonian is given by Eq. (4), is in an oscillatory phase for $0 \leq h \leq 1$. The XXZ model (Eq. (3)) is in the paramagnetic phase for $0 \leq J_z \leq 1$. In general, 2D quantum models are much more challenging to simulate. QubismNet works well on such 2D quantum systems as shown in FIG. 2 (c). With the generalizing width $\delta = 0.4$, we get similar performance compared with the chains, with $\epsilon_t \sim O(10^{-5}) - O(10^{-4})$ and $\epsilon_g \sim O(10^{-3})$.

To demonstrate the finite-size effects, we show in FIG. 3 (a) the ϵ_g against L on the XXZ and XY models ($\delta = 0.6$). The RDM trick is used with $L_b = 8$. The error bars here (and all others in this work) are evaluated by independently and randomly taking the initial values of the variational parameters in the CNN for ten times. The ϵ_g of both models are still slightly decreasing with L for $L > 60$, meaning it is possible to further reduce the errors by taking larger sizes. In the inset of FIG. 3 (a), we fix $L = 64$ and find that ϵ_g well converges as the subsystem size increases to $L_b > 6$ for the models under consideration. Note it would be inefficient to improve the performance by increasing L_b as the complexity will increase exponentially with it. We have tried larger L_b and the results indicate little improvement to the performance.

In FIG. 3 (b), we show ϵ_g versus δ for the QIM, XXZ, and XY models. We fix $L = 64$ and use the RDM trick with $L_b = 8$. Since the QubismNet only learns from the states sampled in $0 < h < 0.5 - \delta/2$ and $0.5 + \delta/2 < h < 1$, it requires more generalization power to estimate the h of the ground states in $0.5 - \delta/2 < h < 0.5 + \delta/2$ as δ increases. Therefore, the generalizing error ϵ_g of the QubismNet monotonously increases with δ . But even for $\delta = 0.8$, the generalizing error is still insignificant with approximately $\epsilon_g < 0.05$. Meanwhile, the estimations become more fluc-

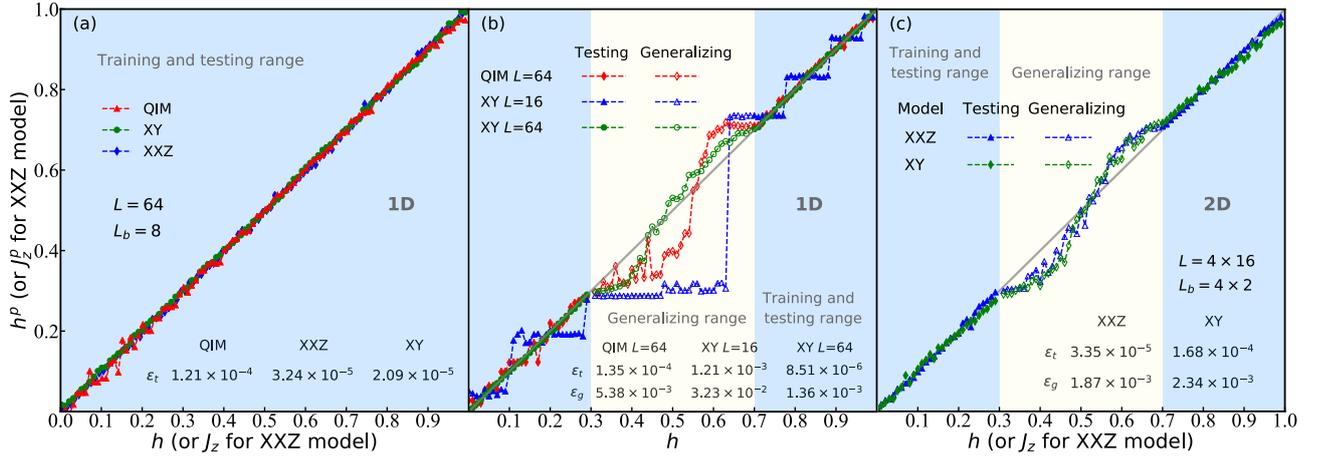


FIG. 2. (Color online) The estimated parameters of the Hamiltonian (h or J_z) versus their ground truth. The blue shadows indicate the ranges of parameters where we take as the training and testing sets. The yellow shadows indicate the range of the generalizing set. The errors of testing and generalizing sets (ϵ_t and ϵ_g) can be found in the inset tables. (a) Estimations h^p or J_z^p versus the true values h or J_z without generalizing set ($\delta = 0$) for the 1D spin models. We take $L = 64$ with the subsystem size $L_b = 8$. The RDM trick is used. (b) Estimations h^p versus the true h taking $\delta = 0.4$ for the 1D spin models. Note the critical point of QIM, $h = 0.5$, is put in the middle of the generalizing range. And no RDM trick is used when $L = 16$ in the XY model. (c) Estimations h^p versus the true h for the spin models on a 4×16 square lattice.

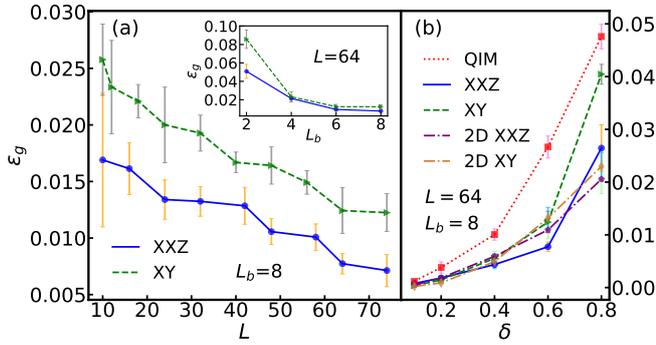


FIG. 3. (Color online) (a) The generalizing error ϵ_g versus the system size L by fixing the generalizing width $\delta = 0.6$ and bulk size $L_b = 8$ for 1D XXZ and XY models. The inset demonstrates the relation between ϵ_g and L_b with $L = 64$. (b) The relation between ϵ_g and δ for 1D and 2D spin models where we fix $L = 64$ and $L_b = 8$.

tuated for larger δ when randomly initializing the variational parameters of the QubismNet.

In above, we stated that after transforming the states into images, we can take advantage of the power of CNN on processing images. Below, we try to directly reshape the coefficients of a RDM into a 2^{2L_b} -dimensional vector. Then we use a 1D version of CNN, which consists of 1D convolutional and pooling layers, to map the vector to the estimations of the target parameters. FIG. 4 (a) shows the estimations h^p versus h on the 1D XY model with $L = 64$ and $L_b = 6$ and 8. The ϵ_t and ϵ_g without the Qubism map becomes more than ten times larger than that with the Qubism map. These results imply that the Qubism map is a reasonable choice, since the image "visualizes" the physics of the state in the patterns of fractals

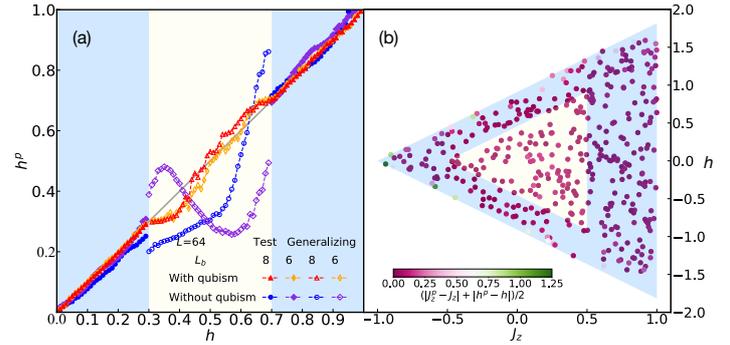


FIG. 4. (Color online) (a) The estimations h^p versus the true h on the 1D XY model ($L = 64$) with and without the Qubism map. We use the RDM trick with $L_b = 8$ and $L_b = 6$. (b) The estimations of h and J_z on the 1D XXZ model. The position of each dot shows the ground truth of the two parameters, and its color indicates the error.

[39]. We do not exclude the possibilities of other maps that may outperform the Qubism map.

The estimation of two parameters is also tried on 1D Heisenberg XXZ model, where both the magnetic field h and J_z which represents the anisotropy are estimated simultaneously. We choose $L = 64$ and use the RDM trick with $L_b = 8$. As shown in FIG. 4 (b), the training and testing set are obtained within the blue region, and the generalizing set are from the yellow region. The ground state is in a gapless phase. The color of each dots illustrates the error between estimation and label. The generalizing error is 7.54×10^{-3} .

SUMMARY AND PERSPECTIVE

Our work is a start-up of using the classical ML models to directly learn the quantum data (e.g., wave-functions or density matrices). CNN models possess high non-linearity, thus it would be interesting to compare with the parameterized quantum circuit models [49–51] that normally represent unitary transformations on quantum states. Our results suggest the impressive learning and generalization powers of CNN in such issues. It could provide a key tool in designing the Hamiltonian in order to, for instance, prepare target states in Hamiltonian-based quantum simulators [38, 52]. An important topic for the future investigations is to test the generalization power while breaking the data balance to different extents. Our proposal can be generalized to learn from the experimental data of quantum measurements in, e.g., a quantum state tomography process [26, 53–55]. The idea of using NN to inversely solve challenging numeric problems can be potentially generalized, e.g., to the constraint satisfaction problems [56, 57].

ACKNOWLEDGMENTS

SJR is grateful to Ding Liu and Ya-Tai Miu for helpful discussions. SJR is supported by NSFC (Grant No. 12004266 and No. 11834014), Beijing Natural Science Foundation (No. 1192005 and No. Z180013), Foundation of Beijing Education Committees (No. KM202010028013), and the Academy for Multidisciplinary Studies, Capital Normal University. Z. C. Tu is supported by the National Natural Science Foundation of China (Grant No. 11975050).

* Corresponding author. Email: sjran@cnu.edu.cn

- [1] F. Verstraete, V. Murg, and J. I. Cirac, Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems, *Advances in Physics* **57**, 143 (2008).
- [2] R. Orús, Tensor networks for complex quantum systems, *Nature Reviews Physics* **1**, 538 (2019).
- [3] S.-J. Ran, E. Tarrico, C. Peng, X. Chen, L. Tagliacozzo, G. Su, and M. Lewenstein, Tensor network contractions: Methods and applications to quantum many-body systems, *Springer Nature* [10.1007/978-3-030-34489-4](https://doi.org/10.1007/978-3-030-34489-4) (2020).
- [4] D. Ceperley and B. Alder, Quantum monte carlo, *Science* **231**, 555 (1986).
- [5] M. P. Nightingale and C. J. Umrigar, Quantum monte carlo methods in physics and chemistry, *Springer Netherlands NATO ASI series: Mathematical and physical sciences* (1998).
- [6] L. Wang, Discovering phase transitions with unsupervised learning, *Physical Review B* **94**, 195105 (2016).
- [7] J. Carrasquilla and R. G. Melko, Machine learning phases of matter, *Nature Physics* **13**, 431 (2017).
- [8] E. P. Van Nieuwenburg, Y.-H. Liu, and S. D. Huber, Learning phase transitions by confusion, *Nature Physics* **13**, 435 (2017).
- [9] P. Zhang, H. Shen, and H. Zhai, Machine learning topological invariants with neural networks, *Physical Review Letters* **120**, 066401 (2018).
- [10] 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, *Nature Physics* **15**, 917 (2019).
- [11] J. F. Rodriguez-Nieva and M. S. Scheurer, Identifying topological order through unsupervised machine learning, *Nature Physics* **15**, 790 (2019).
- [12] M. S. Scheurer and R.-J. Slager, Unsupervised machine learning and band topology, *Physical Review Letters* **124**, 226401 (2020).
- [13] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. Von Lilienfeld, Fast and accurate modeling of molecular atomization energies with machine learning, *Physical Review Letters* **108**, 058301 (2012).
- [14] T. Xie and J. C. Grossman, Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties, *Physical Review Letters* **120**, 145301 (2018).
- [15] P. Z. Hanakata, E. D. Cubuk, D. K. Campbell, and H. S. Park, Accelerated search and design of stretchable graphene kirigami using machine learning, *Physical Review Letters* **121**, 255304 (2018).
- [16] G. Carleo and M. Troyer, Solving the quantum many-body problem with artificial neural networks, *Science* **355**, 602 (2017).
- [17] K. Choo, G. Carleo, N. Regnault, and T. Neupert, Symmetries and many-body excitations with neural-network quantum states, *Physical Review Letters* **121**, 167204 (2018).
- [18] I. Glasser, N. Pancotti, M. August, I. D. Rodriguez, and J. I. Cirac, Neural-network quantum states, string-bond states, and chiral topological states, *Physical Review X* **8**, 011006 (2018).
- [19] D.-L. Deng, X. Li, and S. D. Sarma, Machine learning topological states, *Physical Review B* **96**, 195145 (2017).
- [20] A. Avella, F. Mancini, et al., Strongly correlated systems, Springer [10.1007/978-3-642-35106-8](https://doi.org/10.1007/978-3-642-35106-8) (2012).
- [21] Y. Kuramoto, Quantum many-body physics, Springer Japan [10.1007/978-4-431-55393-9](https://doi.org/10.1007/978-4-431-55393-9) (2020).
- [22] R. Fournier, L. Wang, O. V. Yazyev, and Q. Wu, Artificial neural network approach to the analytic continuation problem, *Physical Review Letters* **124**, 056401 (2020).
- [23] Y. H. Teoh, M. Drygala, R. G. Melko, and R. Islam, Machine learning design of a trapped-ion quantum spin simulator, *Quantum Science and Technology* **5**, 024001 (2020).
- [24] P. Z. Hanakata, E. D. Cubuk, D. K. Campbell, and H. S. Park, Forward and inverse design of kirigami via supervised autoencoder, *Physical Review Research* **2**, 042006 (2020).
- [25] L.-F. Arsenault, R. Neuberger, L. A. Hannah, and A. J. Millis, Projected regression method for solving fredholm integral equations arising in the analytic continuation problem of quantum physics, *Inverse Problems* **33**, 115007 (2017).
- [26] T. Xin, S. Lu, N. Cao, G. Anikeeva, D. Lu, J. Li, G. Long, and B. Zeng, Local-measurement-based quantum state tomography via neural networks, *NPJ Quantum Information* **5**, 1 (2019).
- [27] G. Hegde and R. C. Bowen, Machine-learned approximations to density functional theory hamiltonians, *Scientific Reports* **7**, 42669 (2017).
- [28] X.-Y. Li, F. Lou, X. Gong, and H. Xiang, Constructing realistic effective spin hamiltonians with machine learning approaches, *New Journal of Physics* **22**, 053036 (2020).
- [29] A. Sehanobish, H. H. Corzo, O. Kara, and D. van Dijk, Learning potentials of quantum systems using deep neural networks, arXiv preprint arxiv.org/abs/2006.13297 (2020).

- [30] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, Backpropagation applied to handwritten zip code recognition, *Neural Computation* **1**, 541 (1989).
- [31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, Imagenet classification with deep convolutional neural networks, *Communications of the ACM* **60**, 84 (2017).
- [32] N. Aloysius and M. Geetha, A review on deep convolutional neural networks, *2017 International Conference on Communication and Signal Processing*, 0588 (2017).
- [33] G. Yao, T. Lei, and J. Zhong, A review of convolutional-neural-network-based action recognition, *Pattern Recognition Letters* **118**, 14 (2019).
- [34] F. Sultana, A. Sufian, and P. Dutta, A review of object detection models based on convolutional neural network, *Intelligent Computing: Image Processing Based Applications*, 1 (2020).
- [35] N. F. Berthussen, Y. Sizyuk, M. S. Scheurer, and P. P. Orth, Learning crystal field parameters using convolutional neural networks, arXiv preprint arxiv.org/abs/2011.12911 (2020).
- [36] G. B. Goh, C. Siegel, A. Vishnu, N. O. Hodas, and N. Baker, Chemception: a deep neural network with minimal chemistry knowledge matches the performance of expert-developed qsar/qspr models, arXiv preprint arxiv.org/abs/1706.06689 (2017).
- [37] N. Laanait, J. Romero, J. Yin, M. T. Young, S. Treichler, V. Starchenko, A. Borisevich, A. Sergeev, and M. Matheson, Exascale deep learning for scientific inverse problems, arXiv preprint arxiv.org/abs/1909.11150 (2019).
- [38] A. Das and B. K. Chakrabarti, Quantum annealing and analog quantum computation, *Reviews of Modern Physics* **80**, 1061 (2008).
- [39] J. Rodríguez-Laguna, P. Migdał, M. I. Berganza, M. Lewenstein, and G. Sierra, Qubism: self-similar visualization of many-body wavefunctions, *New Journal of Physics* **14**, 053028 (2012).
- [40] S. R. White, Density matrix formulation for quantum renormalization groups, *Physical Review Letters* **69**, 2863 (1992).
- [41] S. R. White, Density-matrix algorithms for quantum renormalization groups, *Physical Review B* **48**, 10345 (1993).
- [42] G. Hinton, N. Srivastava, and K. Swersky, Neural networks for machine learning lecture 6a overview of mini-batch gradient descent, *Cited on* **14** (2012).
- [43] Note that the datasets, i.e., the ground states of the Hamiltonians with different physical parameters, are prepared by the exact diagonalization algorithm for small sizes or by DMRG algorithm for large sizes.
- [44] F. Verstraete and J. I. Cirac, Matrix product states represent ground states faithfully, *Physical Review B* **73**, 094423 (2006).
- [45] V. Zauner, D. Draxler, L. Vanderstraeten, M. Degroote, J. Haegeman, M. M. Rams, V. Stojevic, N. Schuch, and F. Verstraete, Transfer matrices and excitations with matrix product states, *New Journal of Physics* **17**, 053002 (2015).
- [46] F. Franchini, An introduction to integrable techniques for one-dimensional quantum systems, Springer **940**, 10.1007/978-3-319-48487-7 (2017).
- [47] R. Schaffer, Y. Huh, K. Hwang, and Y. B. Kim, Quantum spin liquid in a breathing kagome lattice, *Physical Review B* **95**, 054410 (2017).
- [48] C. Repellin, Y.-C. He, and F. Pollmann, Stability of the spin-1/2 kagome ground state with breathing anisotropy, *Physical Review B* **96**, 205124 (2017).
- [49] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, Quantum circuit learning, *Physical Review A* **98**, 032309 (2018).
- [50] J.-G. Liu and L. Wang, Differentiable learning of quantum circuit born machines, *Physical Review A* **98**, 062324 (2018).
- [51] D. Zhu, N. M. Linke, M. Benedetti, K. A. Landsman, N. H. Nguyen, C. H. Alderete, A. Perdomo-Ortiz, N. Korda, A. Garfoot, C. Brecque, et al., Training of quantum circuits on a hybrid quantum computer, *Science Advances* **5**, eaaw9918 (2019).
- [52] I. M. Georgescu, S. Ashhab, and F. Nori, Quantum simulation, *Reviews of Modern Physics* **86**, 153 (2014).
- [53] K. Vogel and H. Risken, Determination of quasiprobability distributions in terms of probability distributions for the rotated quadrature phase, *Physical Review A* **40**, 2847 (1989).
- [54] M. Cramer, M. B. Plenio, S. T. Flammia, R. Somma, D. Gross, S. D. Bartlett, O. Landon-Cardinal, D. Poulin, and Y.-K. Liu, Efficient quantum state tomography, *Nature Communications* **1**, 1 (2010).
- [55] B. Lanyon, C. Maier, M. Holzäpfel, T. Baumgratz, C. Hempel, P. Jurcevic, I. Dhand, A. Buyskikh, A. Daley, M. Cramer, et al., Efficient tomography of a quantum many-body system, *Nature Physics* **13**, 1158 (2017).
- [56] S. A. Cook, The complexity of theorem-proving procedures, stoc'71: Proceedings of the third annual acm symposium on theory of computing (1971).
- [57] F. Krzakala and L. Zdeborová, Hiding quiet solutions in random constraint satisfaction problems, *Physical review letters* **102**, 238701 (2009).

SUPPLEMENTARY MATERIAL

In the supplementary material, more details about the Qubism map and CNN are provided, and the images obtained by the Qubism map from various models in 1D and 2D are shown. The detailed descriptions and some supplementary data on applying QubismNet to 2D quantum systems on the square and breathing Kagome lattices are given.

Qubism Map

Consider a quantum system with L spins, denoted by $s = \{X_1 Y_1 X_2 Y_2 \dots X_{L/2} Y_{L/2}\}$ with $X_i, Y_i \in \{0, 1\}$. The resolution of the image obtained by the Qubism map is $(2^{L/2}, 2^{L/2})$. Each spin configuration corresponds to the pixel in the x -th row and y -th column of the image satisfying

$$\begin{aligned} x &= \sum_{i=1}^{L/2} X_i 2^{(L/2-i)} + 1, \\ y &= \sum_{i=1}^{L/2} Y_i 2^{(L/2-i)} + 1. \end{aligned} \quad (5)$$

The gray-scale value of each pixel is taken as the coefficient of the quantum state in the corresponding spin configuration. Besides, each pixel can be attached to colors based on the phase of quantum state. The images obtained by the Qubism map of 1D QIM, Heisenberg XY, and XXZ models are shown in FIG. 5 (a), (b) and (c), respectively. In all these cases, we take $L = 64$ and $L_b = 8$.

Convolutional Neural Network

CNN has shown the advantage in image recognition and many other challenging tasks. In general, CNN consists of several alternative convolutional and pooling layers, which together serve as a feature extractor. One or many fully-connected layers are then used to map the extracted features to the target output in classification and regression. The CNN we use in this work contains eight layers. The first convolutional layer filters the input images with 32 kernels of size 3×3 and a stride of size 1×1 . The second convolutional layer takes the output of the first convolutional layer as input with the same setting as the first convolutional layer. The first max-pooling layer of size 2×2 follows which downsamples the size of features. Then the third and fourth convolutional layers are used both with 64 kernels of size 3×3 and a stride of size 1×1 . All convolutional layers use padding around the images so the outputs have the same height/width dimension as the inputs. The second max-pooling layer follows with pool size 2×2 . Next, the output of the second max-pooling layer is flattened and then input to two fully-connected layers with 128 and 32 neurons respectively. The output of the last fully connected layer is fed to one neuron which produces the final output of

the physical parameter. Note that we choose two successive convolutional layers to improve the quality of features because of the more complex nonlinearity and larger receptive fields. The rectified linear unit (ReLU) [1] is chosen as activation function for all the convolutional and fully-connected layers. A linear activation function is used for the output layer. The optimizer we use is RMSprop with the initial learning rate 0.001. "He normal initialization" is employed as initialization method which has been proven to get good effect in CNN with the rectifier nonlinearities [2].

We split data into four datasets which are training, validation, testing and generalizing set. The training set is used to train the CNN model. The validation set is randomly taken from the training set with the percentage of 10%. The validation set has two roles in our work. One is choosing the proper hyper-parameters such as the number of neurons in each layer, the optimizer, and so on. The other is the early-stopping technique. The testing set and the generalizing set consist of data which never show in the training set. The difference is that data in the testing set have the same range of the training set however the generalizing range is beyond the training range.

Two tricks on avoiding overfitting during the training process are employed. One is the dropout [3, 4] with rate of $p = 0.5$, meaning randomly masking 50% of neurons during training. It is a commonly used method making the NN robust. The other is validation. We always save the best model selected by the minimum loss of the validation set which consists of the samples randomly chosen 10% from the training set. The best model is then used on the testing and generalizing set. Note that the input images are normalized so that the features are in $[0, 1]$. We construct and train our CNN with Keras [5], a high-level open-source API running on top of TensorFlow. We train the CNN for 300 epochs for each case (one epoch means training by all the samples in training set once). The total training time is roughly 0.3 hours on a server equipped with NVIDIA P100 graphics processing unit when we use 1000 training samples.

FIG. 6 (a) shows the loss functions versus epochs for the 1D QIM and XY models with $L = 64$, $L_b = 8$ and $\delta = 0.4$. The loss functions of the training and validation sets decrease rapidly and become quite small in a few epochs for all the three models. This indicates that QubismNet is trainable and feasible. Besides, the validation loss functions do not increase during the whole training process implying no overfitting.

Furthermore, we go deep inside the CNN and see extracted features by the convolutional/pooling part of the CNN. After the last pooling layer, each sample is mapped to a $64 \times 64 \times 64$ tensor as the extracted features where 64 is the number of channels. FIG. 6 (b) shows the average magnitude of each channel c . We find the dominant contribution is from the first channel in almost all the cases. In FIG. 6 (c), we demonstrate the 64×64 features in the first channel. Interestingly, we observe that the prominent features (with larger values marked by the deeper dots) move in general from right to left with the increase of h . These results indicate that QubismNet captures some consecutive rule from the quantum states in its own way.

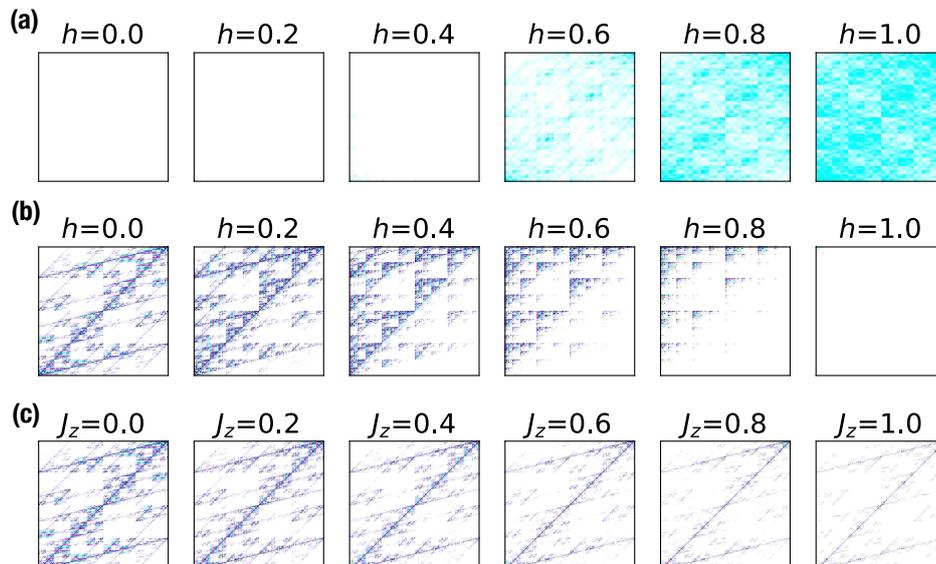


FIG. 5. (Color online) Illustrations of the images by the Qubism map from the ground states of the QIM, XY, and XXZ models.

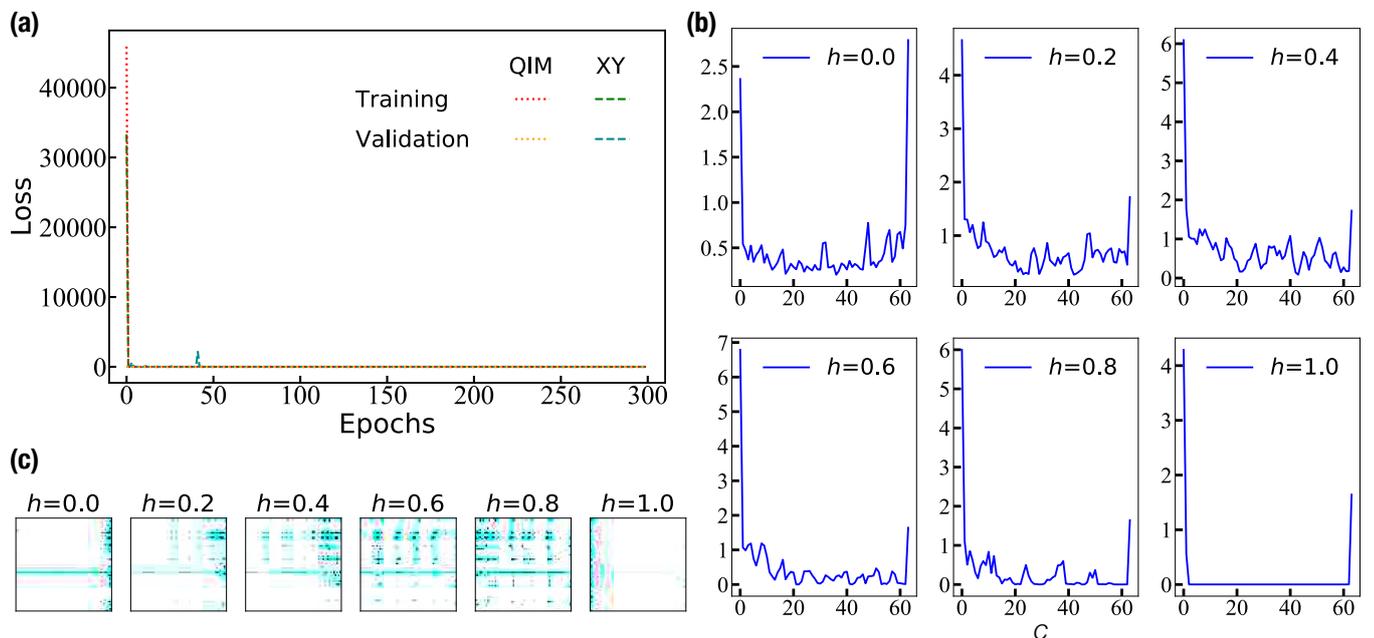


FIG. 6. (Color online) The loss functions and extracted features of the CNN. (a) The decreases of the loss function with the training epochs on training and validation set for QIM, XY and XXZ models. (b) For the XY model, we show the average magnitude of the extracted features for different channels c . For different h , the dominant contribution is always from the first channel. (c) The extracted 64×64 features in the first channel. In general, we observe that the prominent features (illustrated by the dark dots) move from right to left as h increases.

All codes used in the manuscript and this supplementary material can be found on GitHub [6].

RDM-based method for two-dimensional lattice

We follow the standard recipe of DMRG on solving the ground states of 2D quantum models [7]. The 2D lattice model with nearest-neighbor interactions is stretched into a chain with long-range interactions, as illustrated in FIG. 7 (a). We set the size of lattice as 4×16 for both the 2D XY and

XXZ models. The subsystem used to calculate RDM is chosen in the middle of lattice with size 4×2 as illustrated. For the purified state of ρ^2 , we treat all the degrees of freedom in the *bra* space as one of the two dimensions of the 2D image, and treat those in the *ket* space as the other dimension of the image. This ordering is illustrated in FIG. 7 (b). Several images obtained from the ground states of the 2D XY and XXZ models with different h or J_z are shown in FIG. 7 (c) and (d) as examples.

QubismNet for breathing kagome antiferromagnet

We also tried QubismNet for the non-trivial breathing kagome antiferromagnet with the Hamiltonian

$$\begin{aligned} \hat{H} = & J_{\Delta} \sum_{\langle i,j \rangle \in \Delta} (\hat{S}_i^x \hat{S}_j^x + \hat{S}_i^y \hat{S}_j^y + \hat{S}_i^z \hat{S}_j^z) \\ & + J_{\nabla} \sum_{\langle i,j \rangle \in \nabla} (\hat{S}_i^x \hat{S}_j^x + \hat{S}_i^y \hat{S}_j^y + \hat{S}_i^z \hat{S}_j^z) \end{aligned} \quad (6)$$

where J_{Δ} and J_{∇} represent the coupling strength in the up and down triangles, respectively. The illustration of the kagome lattice with periodic boundary condition is shown in FIG. 8 (a). The number of columns is three and the image size after the qubism map is 512×512 . The breathing anisotropy J_{∇}/J_{Δ} is chosen to vary between 0.4 and 1.0. The ground

states are magnetically disordered RVB states and solved by DMRG algorithm. FIG. 8 (b) shows the results of predicting the breathing anisotropy on the testing and generalizing sets. The mean square errors are of the order of magnitude $O(10^{-4})$. This demonstrates the validity of the QubismNet on the nontrivial kagome model with magnetically disordered ground states.

* Corresponding author. Email: sjran@cnu.edu.cn

- [1] V. Nair and G. E. Hinton, Rectified linear units improve restricted boltzmann machines, in [ICML](#) (2010) pp. 807–814.
- [2] K. He, X. Zhang, S. Ren, and J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, [Proceedings of the IEEE international conference on computer vision](#), 1026 (2015).
- [3] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, arXiv preprint [arXiv:1207.0580](https://arxiv.org/abs/1207.0580) arxiv.org/abs/1207.0580 (2012).
- [4] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, [Journal of Machine Learning Research](#) **15**, 1929 (2014).
- [5] F. Chollet et al., Keras, <https://keras.io> (2015).
- [6] <https://github.com/JoyeBNU/QubismNet>.
- [7] E. Stoudenmire and S. R. White, Studying two-dimensional systems with the density matrix renormalization group, [Annual Review of Condensed Matter Physics](#) **3**, 111 (2012).

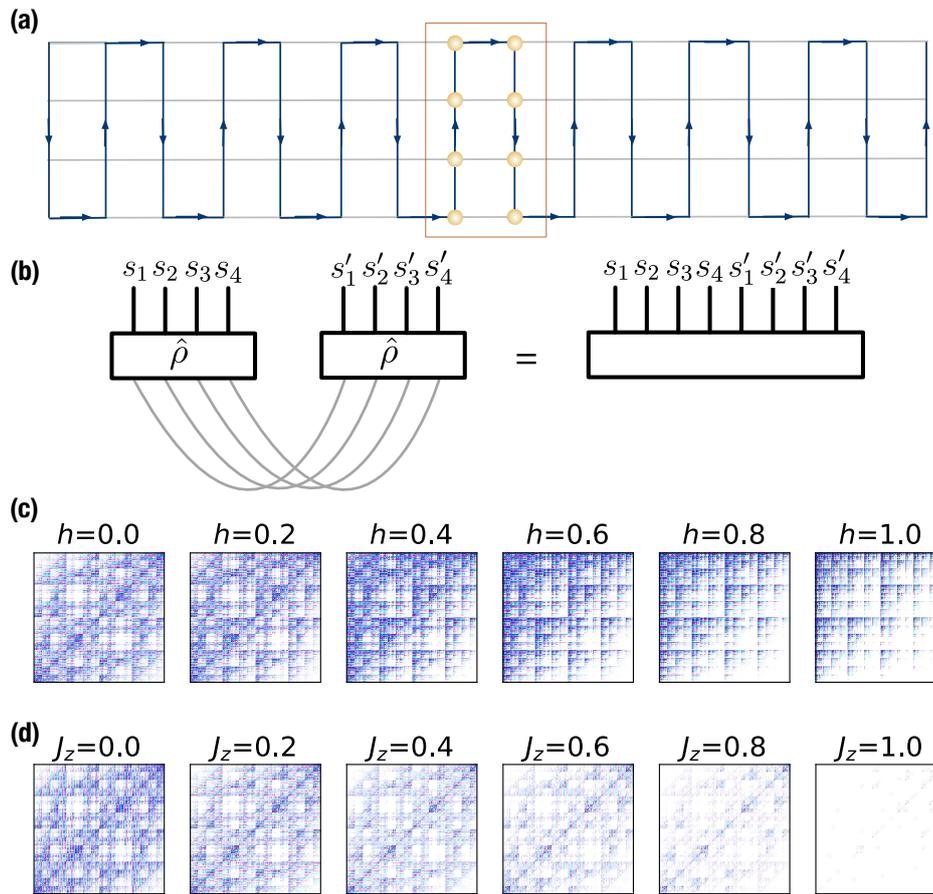


FIG. 7. (Color online) An illustration of DMRG applied to the 2D system, and the images obtained from the 2D ground states. (a) An illustration of how the 2D lattice is stretched to a 1D chain in order to use DMRG to simulate the ground states. We choose a 2×4 sub-system in the middle to define the RDM. (b) An illustration of how the reduced density matrices are constructed. (c) The images obtained by applying the Qubism map to the ground states in the 2D XX model in different transverse fields h . (d) The images from the 2D XXZ model with different J_z .

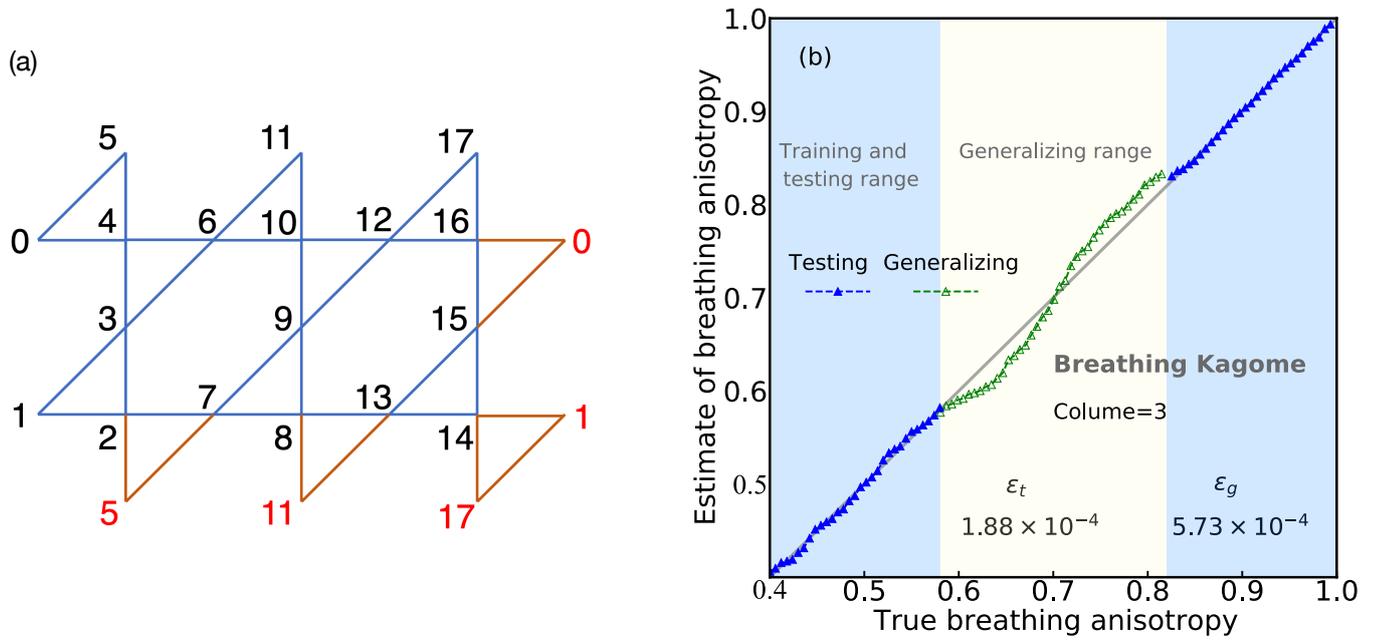


FIG. 8. (Color online) (a) An illustration of the breathing kagome antiferromagnet, where the couplings of the upper and lower triangles are antisymmetric. The number of columns is three. Periodic boundary condition is chosen as shown in the red edges. (b) shows the estimations of the breathing anisotropy versus its ground truth. We use the RDM trick with $L_b = 8$.