\



Dynamical learning of non-Markovian quantum dynamics

Jintao Yang(杨锦涛), Junpeng Cao(曹俊鹏), and Wen-Li Yang(杨文力)
Citation: Chin. Phys. B, 2022, 31 (**1**): 010314. DOI: 10.1088/1674-1056/ac2490
Journal homepage: http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

Low-loss belief propagation decoder with Tanner graph in quantum error-correction codes

Dan-Dan Yan(颜丹丹), Xing-Kui Fan(范兴奎), Zhen-Yu Chen(陈祯羽), and Hong-Yang Ma(马鸿洋)†
Chin. Phys. B, 2022, 31 (**1**): 010304. DOI: 10.1088/1674-1056/ac11cf

Determination of quantum toric error correction code threshold using convolutional neural network decoders

Hao-Wen Wang(王浩文), Yun-Jia Xue(薛韵佳), Yu-Lin Ma(马玉林), Nan Hua(华南), and Hong-Yang Ma(马鸿洋)
Chin. Phys. B, 2022, 31 (**1**): 010303. DOI: 10.1088/1674-1056/ac11e3

Quantum annealing for semi-supervised learning

Yu-Lin Zheng(郑玉鳞), Wen Zhang(张文), Cheng Zhou(周诚), and Wei Geng(耿巍)
Chin. Phys. B, 2021, 30 (**4**): 040306. DOI: 10.1088/1674-1056/abe298

Experimental implementation of a continuous-time quantum random walk on a solid-state quantum information processor

Maimaitiyiming Tusun(麦麦提依明·吐孙), Yang Wu(伍旸), Wenquan Liu(刘文权), Xing Rong(荣星), Jiangfeng Du(杜江峰)
Chin. Phys. B, 2019, 28 (**11**): 110302. DOI: 10.1088/1674-1056/ab44ae

Nuclear magnetic resonance for quantum computing: Techniques and recent achievements

Tao Xin(辛涛), Bi-Xue Wang(王碧雪), Ke-Ren Li(李可仁), Xiang-Yu Kong(孔祥宇), Shi-Jie Wei(魏世杰), Tao Wang(王涛), Dong Ruan(阮东), Gui-Lu Long(龙桂鲁)
Chin. Phys. B, 2018, 27 (**2**): 020308. DOI: 10.1088/1674-1056/27/2/020308

Dynamical learning of non-Markovian quantum dynamics

Jintao Yang(杨锦涛)[1,2,†], Junpeng Cao(曹俊鹏)[1,2,3,4,‡], and Wen-Li Yang(杨文力)[4,5,6,7,§]

[1] *Beijing National Laboratory for Condensed Matter Physics, Institute of Physics, Chinese Academy of Sciences, Beijing* 100190, *China*
[2] *School of Physical Sciences, University of Chinese Academy of Sciences, Beijing* 100049, *China*
[3] *Songshan Lake Materials Laboratory, Dongguan* 523808, *China*
[4] *Peng Huanwu Center for Fundamental Theory, Xi'an* 710127, *China*
[5] *Institute of Modern Physics, Northwest University, Xi'an* 710127, *China*
[6] *School of Physical Sciences, Northwest University, Xi'an* 710127, *China*
[7] *Shaanxi Key Laboratory for Theoretical Physics Frontiers, Xi'an* 710127, *China*

(Received 3 June 2021; revised manuscript received 19 July 2021; accepted manuscript online 8 September 2021)

We study the non-Markovian dynamics of an open quantum system with machine learning. The observable physical quantities and their evolutions are generated by using the neural network. After the pre-training is completed, we fix the weights in the subsequent processes thus do not need the further gradient feedback. We find that the dynamical properties of physical quantities obtained by the dynamical learning are better than those obtained by the learning of Hamiltonian and time evolution operator. The dynamical learning can be applied to other quantum many-body systems, non-equilibrium statistics and random processes.

Keywords: machine learning, quantum dynamics, open quantum system

PACS: 03.65.Yz, 47.10.Fg, 02.50.Ga **DOI:** 10.1088/1674-1056/ac2490

1. Introduction

Machine learning is a very important technique in the study of modern physics, mathematics, chemistry, and medicine. Artificial neural network is a powerful tool to realize the machine learning. Focus on the physics, artificial neural networks are used to study phase transitions and critical phenomena.[1–13] topological nature of quantum states,[14–18] many-body correlated effects,[19–26] optimization of numerical simulations,[27–29] quantum error correction codes,[30–33] and so on.[34–37]

Recently, machine learning is also applied to the open quantum systems which have many applications in various fields such as solid state physics, quantum chemistry, quantum sensing, quantum information transport, and quantum computing. Luchnikov *et al.* applied the supervised learning to the open quantum system.[38] By constructing the likelihood function of a specific sequence of physical quantity and using it as a loss function for back-propagation, they simulated the model Hamiltonian of an open quantum system.

Neural network has been used in the machine leaning extensively. By adjusting the connection weights among neurons in the neural network, machine learning can be carried out. Later, the neurobiology experiments show that the static weights can also be used to realize the fast learning.[39]

In this paper, we study the non-Markovian dynamics of an open quantum system by constructing a biological recurrent neural network. With the help of stored data library, all the weights after appropriate pre-training are fixed. Thus the weights learning is not required and the dynamical properties can be learned quickly.[40] Then we generate the dynamical behaviors of physical quantities by this kind of supervised dynamical learning.

There are two ways to carry out the machine learning. One is simulating the system Hamiltonian. We show that only if the Hamiltonian is generated accurately, the corresponding dynamic behaviors can be true. The other is the dynamical learning of time evolution of the physical quantities. We find that the neural network with static weights can give the dynamic properties very well.

2. Non-Markovian quantum dynamics

Quantum dynamics is the result of interactions between the system and the environment. Generally, the environment is complicated and can affect the dynamical behavior of the system, thus can be regarded as an effective reservoir. From the dynamical properties of the system, some information about the environment could be obtained. The contributions of environment are mainly attributed to the memory effects accompanying non-Markovian dynamics.

When measuring a quantum system, if the degree of freedom of effective reservoir is equal to one, the system evolution is Markovian. That is to say, the measurement results at current moment are only related to the measurement values at

†Corresponding author. E-mail: jintao_yang@iphy.ac.cn
‡Corresponding author. E-mail: junpengcao@iphy.ac.cn
§Corresponding author. E-mail: wlyang@nwu.edu.cn

© 2022 Chinese Physical Society and IOP Publishing Ltd

previous moment. Thus the present results do not depend on the overall measurement results in the earlier period of time.

In this paper, we consider the non-Markovian quantum dynamics and the system is described as[38]

$$\begin{aligned}&\rho_{\rm S}(t)=\mathrm{tr}_{\rm ER}[\rho_{\rm S+ER}(t)],\\&\frac{\mathrm{d}\rho_{\rm S+ER}(t)}{\mathrm{d}t}=L_{\rm S+ER}[\rho_{\rm S+ER}(t)],\end{aligned}\tag{1}$$

where $\rho_{\rm S}(t)$ is the density operator of the system, $\rho_{\rm S+ER}(t)$ is the density operator of system and environment, and $L_{\rm S+ER}$ is the generator of non-Markovian process. In system (1), the environment is divided into two parts. The first part carries the memory and is responsible for non-Markov dynamics. The second part does not have the memory and leads to the Markovian dissipation and decoherence between the system and the effective reservoir.

Now, we carry out the machine learning. The density operator $\rho_{\rm S}(t)$ can not be achieved by the machine through a single learning. Only through many times of learning, the relevant information about the system can be obtained. The learning means the measurement, that is acting a projection operator on the system and the effective reservoir. Considering a series of projection measurements with different time, the measurement results seem useless, because the wave function of quantum system collapses with each measurement. However, such series of measurement results do contain some necessary information related with the system. It is proved that an enormous number of measurements has a pattern which can be recognized by the machine.[38] Then we can extract the effective information about the system from the corresponding pattern.

In order to learn the non-Markovian dynamics, we first need to construct a dataset for machine learning. We arrange the projection measurements in the order of time. In the process of obtaining training data, the time evolution rule is given by the generator $L_{\rm S+ER}$.

The dynamical properties of the system are quantified by the time evolution operator $U(t)=\mathrm{e}^{-\mathrm{i}Ht/\hbar}$, where H is the Hamiltonian. The time evolution of a state satisfies $|\psi(t)\rangle=U(t)|\psi(0)\rangle$, where $|\psi(0)\rangle$ is the initial state. Then the density operator reads $\rho_{\rm S}(t)=|\psi(t)\rangle\langle\psi(t)|$. We note that the time evolution operator can also generate the training dataset.

For simplicity, we choose the observable physical quantities as the expected values of Pauli matrices

$$\langle\boldsymbol{\sigma}\rangle=\langle\psi(t)|\boldsymbol{\sigma}|\psi(t)\rangle,\tag{2}$$

and study their dynamical properties. It is clear that the expected values depend on the time evolution operator thus the Hamiltonian. Therefore, in order to obtain the dynamical properties of these physical quantities, we should learn the Hamiltonian first. Here we simulate the Hamiltonian by using the supervised learning. Obviously, the results are better when the simulated Hamiltonian is closer to the actual values.

3. Learning of Hamiltonian

In this section, we start from the Hamiltonian of a given quantum system to show the machine learning results. We randomly generate a Hamiltonian as

$$H=\frac{1}{2}W\otimes I,\tag{3}$$

where W is a $d^2\times d^2$ Hermitian matrix and I is the $d\times d$ identity matrix. Thus the Hamiltonian is a $d^3\times d^3$ matrix. For simplicity, we put $d=2$ and the matrix elements of W take the random complexes with the absolute values of both real and imaginary parts less than one. The Hamiltonian in Eq. (3) includes the system and the environment. The dimension of the system is d. The environment is divided into two parts and the dimension of each part is d. The W in Eq. (3) corresponds to the combination of the system and the first part of the environment. The first part carries the memory and is responsible for the non-Markov dynamics. The second part does not have the memory and leads to the Markovian dissipation. For simplicity, we set the second part of the environment as the identity matrix.

We first learn the Hamiltonian (3) by using the supervised learning. The main idea of supervised learning is as follows. During the learning, we define a loss function $p(\{E_j\}_{j=1}^N|H_i)$, where E_j is the projection operators acting on the system and the effective reservoir, N is the total number of projectors with different time and H_i is the current Hamiltonian. The set of projectors $\{E_j\}_{j=1}^N$ supplies the training set for the machine learning. The loss function is used to quantify the difference between the learned Hamiltonian and the real one. During the training, the learned Hamiltonian changes with each steps. The loss function can be increased as much as possible through the gradient ascent. When the log likelihood $(1/N)\log p(\{E_j\}_{j=1}^N|H_{\rm s})$ of the trained system $H_{\rm s}$ is equal to the actual value $(1/N)\log p(\{E_j\}_{j=1}^N|H)$ which is obtained by the given Hamiltonian (3), we expect that the learned Hamiltonian $H_{\rm s}$ equals to the real one (3) and the dynamical evolution of the physical quantities given by $H_{\rm s}$ meet the actual values.

The log likelihood of supervised learning result and the actual value are shown in Fig. 1(a). We see that after completing the training, the log likelihood of the trained system $H_{\rm s}$ equals that of the real system with the epochs steps larger than 75. Then we calculate the physical quantities $\langle\boldsymbol{\sigma}\rangle$ and the results are shown in Figs. 1(b)–1(d). Here, the blue lines denote the supervised learning results while the red lines denote the actual values which are obtained form Eqs. (2) and

(3). We find that the dynamical evolution of physical quantities obtained from the trained system H_s are slightly different from the actual values, although the log likelihood of trained system is the same as the actual value.

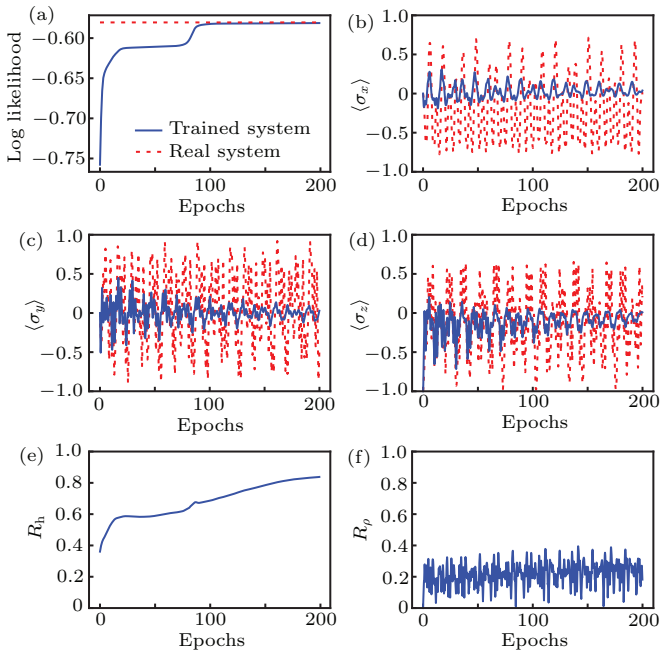


Fig. 1. The supervised learning results of an open quantum system. (a) The log likelihoods of trained system and that of the real system. These two curves converge if the training number is large enough. (b)–(d) The dynamical evolution of physical quantities $\langle\boldsymbol{\sigma}\rangle$. The blue lines are the learned results and the red lines are the actual values. We see that they do not coincide with other. With the increasing of evolution time, the difference becomes larger and larger. (e) The difference between the learned Hamiltonian and the real one. We see that with the increasing of training time, the difference does not decrease. (f) The difference of density operators between the learned one and the real one.

In order to explain above conflict and check the validity of machine learning, we define

$$R_{\rm h} = \frac{1}{N^2}\sum_{m,n}^{N} |H_{mn} - (H_{\rm s})_{mn}|, \tag{4}$$

where the subscript mn means the matrix elements. The quantity $R_{\rm h}$ gives a measure of the difference between the real Hamiltonian H and the learned one $H_{\rm s}$. We also define a measure of the difference of density matrices

$$R_\rho = \frac{1}{N^2}\sum_{m,n}^{N} |(\rho_{\rm S})_{mn} - (\rho_{\rm s})_{mn}|, \tag{5}$$

where $\rho_{\rm S}$ is the density matrix of real system, $\rho_{\rm s}$ is trained one with the definition $\rho_{\rm s} = |\psi_{\rm s}(t)\rangle\langle\psi_{\rm s}(t)|$ and $|\psi_{\rm s}(t)\rangle = {\rm e}^{-{\rm i}H_{\rm s}t/\hbar}|\psi(0)\rangle$. The values of $R_{\rm h}$ and R_ρ are shown in Figs. 1(e)–1(f), respectively. We find that with the increasing of epoch step, the $R_{\rm h}$ increases and the final value is about 0.8. Meanwhile, the value of R_ρ oscillates between 0 and 0.4.

Next, we seek the benchmark which can give the believable results of the dynamical evolution. For this purpose, we construct a trying Hamiltonian

$$H_{\rm t} = H_0 + qV, \tag{6}$$

where q is a small quantity and V is a 8×8 random Hermitian matrix. We note that the generation rules of V and W are the same except for the dimensions. H_0 is generated by the same rule Eq. (3) as the real Hamiltonian. Substituting the trying Hamiltonian $H_{\rm t}$ into Eq. (2), we obtain the trying results of dynamical evolution of the physical quantities $\langle\boldsymbol{\sigma}\rangle$, which are shown in Fig. 2 as the blue lines with different q. Here the red lines represent the actual values. We find that only when q is small enough, the trying results agree with the actual ones. Then we can extract the correct information about the system from the correlation patterns in the series of measurements of the trying Hamiltonian $H_{\rm t}$. Meanwhile, replacing the $H_{\rm s}$ in the measure (4) by the $H_{\rm t}$, we obtain that if the machine learning works, the $R_{\rm h}$ should be very small. Figure 2(e) gives the benchmark as $R_{\rm h} = 1.02\times 10^{-5}$ and figure 2(f) gives the benchmark as $R_{\rm h} = 5.2\times 10^{-5}$. All these values are smaller than that obtained from the learned Hamiltonian $H_{\rm s}$. That is the reason why the learned Hamiltonian can not give the correct dynamical properties of the system.

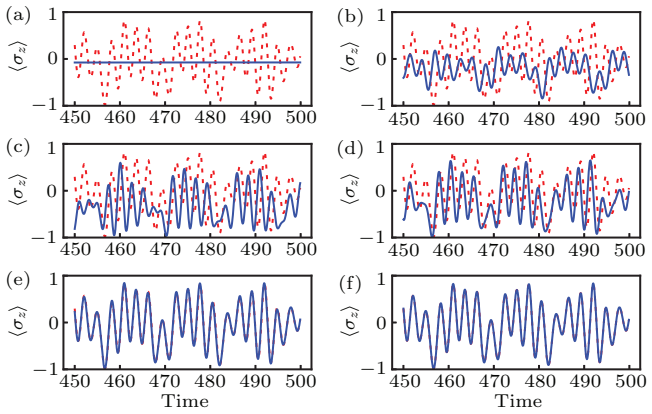


Fig. 2. Dynamical evolution of the physical quantity $\langle\sigma_z\rangle$. The red lines are the actual values and the blue lines are the results obtained from the trying Hamiltonian $H_{\rm t}$ with different q. The values of q in panels (a)–(f) are 1×10^{-1}, 1×10^{-2}, 1×10^{-3}, 5×10^{-4}, 1×10^{-4}, and 5×10^{-5}, respectively. We see that only when q is small enough, the trying results obtained from $H_{\rm t}$ can give the correct dynamical properties of the system. The numerical data of panel (e) give the benchmark as $R_{\rm h} = 1.02\times 10^{-5}$ and that of panel (f) give the benchmark as $R_{\rm h} = 5.2\times 10^{-5}$. Therefore, if the learned Hamiltonian gives the believable results, the $R_{\rm h}$ should be small enough.

In order to solve this problem, we use an alternative method which is the dynamic evolution of physical quantities can be learned directly from the dynamical learning, as shown in the following section.

4. Dynamical learning

The expected values of Pauli matrices can also be expressed as

$$\langle\boldsymbol{\sigma}\rangle = {\rm tr}[\rho_{\rm S}(t)\boldsymbol{\sigma}], \tag{7}$$

which means that the dynamical properties of physical quantities can also be obtained from the density operator. Thus

we carry out the dynamical learning of the density operator directly.

We use a recurrent neural network to realize the dynamical learning, which is a modified version of the echo state network.[39] The neural network constructed in this paper combines the input layer and the output layer to form a loop.[40] Besides the traditional input, our neural network also has a context input, which is used to stabilize the neural network.

The neural network is characterized by

$$\begin{aligned} \dot{x}(t) &= -x(t) + Ar(t) + w_\varepsilon \varepsilon(t) + w_\rho \rho(t) + w_{\rm h} h(t), \\ \rho(t) &= o_\rho r(t), \quad h(t) = o_{\rm h} r(t), \end{aligned} \tag{8}$$

where $x(t)$ is a function of t, A is the recurrent weight of the function $r(t)$ which is related with $x(t)$, w_ε is input weight of input function $\varepsilon(t)$, w_ρ is the feed back weight of the density operator $\rho(t)$, $w_{\rm h}$ is the feed back weight of the context $h(t)$, o_ρ and $o_{\rm h}$ are the output weights of density operator and training context, respectively.

We choose $r(t) = \tanh[x(t) + b]$ and start from an initial uniform distribution. Thus the initial network parameters w_ε, w_ρ, $w_{\rm h}$, and b are determined by the given initial distribution. The recurrent weight A is a matrix whose elements take the values from 0 to 1. During the learning, only the output weights o_ρ and $o_{\rm h}$ are updated via gradient descent method. The initial values of o_ρ, $o_{\rm h}$ are set to zero.

The construction of neural network with dynamical learning includes two steps. The first step is training the neural network. Through training, the neural network will gradually learn how to simulate the dynamic evolution of physical quantities. The second step is testing the network. In this step, the dynamic evolution of physical quantities can be simulated. We note that both of these two steps include the pre-training stage and the dynamical learning stage.

We first train the neural network. We randomly generalize 1000 context matrices $\{\tilde{h}(t)\}$ which have the same structure as Hamiltonian (3). The purpose of this choice is to stabilize the neural network. From these contexts, we obtain the density operators $\{\tilde{\rho}(t)\}$. The context and the density operators in above sequences are put into the neural network in turn to obtain the training set.

In the pre-training of recurrent neural network, the context and density matrix over a period of time are generated by the continuous measurements of input $\tilde{h}(t)$ and $\tilde{\rho}(t)$. We take the difference between the trained density matrix $\rho(t)$ and the input one $\tilde{\rho}(t)$ as the loss function, $\varepsilon(t) = \rho(t) - \tilde{\rho}(t)$, to pre-train the neural network. Meanwhile, we use the FORCE rule[41] for online learning to minimize the loss function $\varepsilon(t)$ through gradient descent method and to update the output wights o_ρ and $o_{\rm h}$ in the neural network.

After the pre-training, we fix the output weights o_ρ and $o_{\rm h}$ of the recurrent neural network. The context $h(t)$ in the neural network is also fixed as a constant, which is the average value of all the previously generated contexts in the pre-training process. Next we carry out the dynamical learning. In this stage, we no longer minimize the loss function $\varepsilon(t)$ and the weights in the neural network are no longer updated. The network generate the trained density operator and trained context.

From the 1000 training, the output weights o_ρ and $o_{\rm h}$ of the neural network are optimized. The trained density operator tends to the actual value of input one. Because the weights of neural network are fixed, the amount of computation are effectively reduced.

Next, we test the neural network and study the dynamical learning results. Substituting the Hamiltonian (3) in to the neural network and repeating the above pre-training and dynamical learning processes, we obtain the learned result of density operator of the system (3). Based on it, we obtain the dynamical learning results of the physical quantities $\langle \boldsymbol{\sigma} \rangle$, which are shown in Figs. 3(a)–3(c). We find that the learned dynamical evolution of physical quantities $\langle \boldsymbol{\sigma} \rangle$ (blue lines) coincide with the actual values (red lines) very well. Meanwhile, the density operator generated by the neural network is very close to the real one, as shown in Fig. 3(d). Therefore, we conclude that the dynamical properties of the non-Markovian quantum systems can be learned by the neural network with fixed weights.

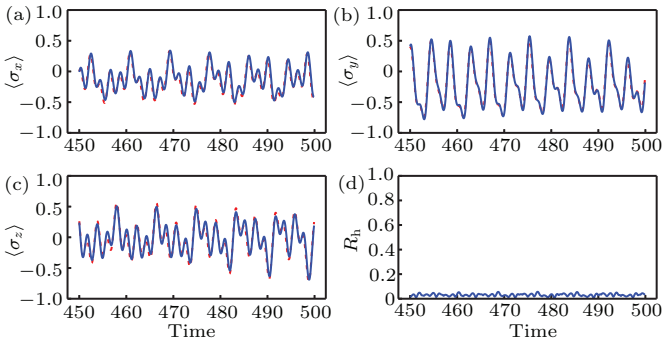


Fig. 3. (a)–(c) The dynamical evolution of physical quantities $\langle \boldsymbol{\sigma} \rangle$. The red lines are the exact dynamics results and the blue lines are the learned results. We see that these curves coincide with each other very well. Thus the neural network with fixed weights can learn the quantum dynamics. (d) The difference between the learned density operator and the actual value. We see that the difference is very small.

Last, we should note that the Hamiltonian is randomly generated according to the rule (3). The neural network constructed in this paper has the ability to accurately learn other quantum systems including the high dimension cases only if they have the same structure as Eq. (3).

5. Conclusion

We study the dynamical learning of Non-Markovian quantum dynamics by using the neural network with fixed

weights. From the generating function of Hamiltonian, we obtain the suitable set of input contexts and the stable neural network. After training, the neural network can generate the learned density operator of a given system very accurately. Based on it, we calculate the dynamical evolution of physical quantities. Comparing with the supervised learning of model Hamiltonian, the dynamical learning results are more accurate.

Dynamic learning has many distinct advantages. For example, it does not need to update the weights of the network during the dynamical learning stage. Thus the amount of computation can be effectively reduced. It avoids the searching of time evolution rule. In some cases, we do not know the Hamiltonian of a dynamical problem. The dynamical learning simulates the dynamical evolution of observable physical quantities directly and the accuracy is very high. We expect that the dynamical learning can be applied to other problems.

Acknowledgements

Project supported by the National Program for Basic Research of the Ministry of Science and Technology of China (Grant Nos. 2016YFA0300600 and 2016YFA0302104), the National Natural Science Foundation of China (Grant Nos. 12074410, 12047502, 11934015, 11975183, 11947301, 11774397, 11775178, and 11775177), the Major Basic Research Program of the Natural Science of Shaanxi Province, China (Grant No. 2017ZDJC-32), the Australian Research Council (Grant No. DP 190101529), the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No. XDB33000000), and the Double First-Class University Construction Project of Northwest University.

References

[1] Ohtsuki T and Ohtsuki T 2016 *J. Phys. Soc. Jpn.* **85** 123706
[2] Carrasquilla J and Melko R G 2017 *Nat. Phys.* **13** 431
[3] Zhang Y and Kim E A 2017 *Phys. Rev. Lett.* **118** 216401
[4] Ch'ng K, Carrasquilla J, Melko R G and Khatami E 2017 *Phys. Rev. X* **7** 031038
[5] Beach M J S, Golubeva A and Melko R G 2018 *Phys. Rev. B* **97** 045207
[6] Rem B S, Kaming N, Tarnowski M, Asteria L, Flaschner N, Becker C, Sengstock K and Weitenberg C 2019 *Nat. Phys.* **15** 917
[7] Huembeli P, Dauphin A and Wittek P 2018 *Phys. Rev. B* **97** 134109
[8] Casert C, Vieijra T, Nys J and Ryckebusch J 2019 *Phys. Rev. E* **99** 023304
[9] Zhang W, Liu J and Wei T C 2019 *Phys. Rev. E* **99** 032142
[10] Huembeli P, Dauphin A, Wittek P and Gogolin C 2019 *Phys. Rev. B* **99** 104106
[11] Kharkov Y A, Sotskov V E, Karazeev A A, Kiktenko E O and Fedorov A K 2020 *Phys. Rev. B* **101** 064406
[12] Sun L W, Li H, Wang P J, Gao H B and Luo M B 2019 *Acta Phys. Sin.* **68** 200701 (in Chinese)
[13] Su Z X, Kang Y Z, Zhang B F, Zhang Z Q and Jiang H 2019 *Chin. Phys. B* **28** 117301
[14] Zhang P, Shen H and Zhai H 2018 *Phys. Rev. Lett.* **120** 066401
[15] Sun N, Yi J, Zhang P, Shen H and Zhai H 2018 *Phys. Rev. B* **98** 085402
[16] Carvalho D, Garcia-Martinez N A, Lado J L and Fernandez-Rossier J 2018 *Phys. Rev. B* **97** 115453
[17] Mano T and Ohtsuki T 2019 *J. Phys. Soc. Jpn.* **88** 123704
[18] Ming Y, Lin C, Bartlett S D and Zhang W W 2019 *npj Comput. Mater.* **5** 88
[19] Carleo G and Troyer M 2017 *Science* **355** 602
[20] Deng D L, Li X and Das Sarma S 2017 *Phys. Rev. X* **7** 021021
[21] Gao X and Duan L M 2017 *Nat. Commun.* **8** 662
[22] Deng D L, Li X and Das Sarma S 2017 *Phys. Rev. B* **96** 195145
[23] Nomura Y, Darmawan A S, Yamaji Y and Imada M 2017 *Phys. Rev. B* **96** 205152
[24] Kaubruegger R, Pastori L and Budich J C 2018 *Phys. Rev. B* **97** 195136
[25] Pastori L, Kaubruegger R and Budich J C 2019 *Phys. Rev. B* **99** 165123
[26] Levine Y, Sharir O, Cohen N and Shashua A 2019 *Phys. Rev. Lett.* **122** 065301
[27] Broecker P, Carrasquilla J, Melko R G and Trebst S 2017 *Sci. Rep.* **7** 8823
[28] Bukov M, Day A G R, Sels D, Weinberg P, Polkovnikov A and Mehta P 2018 *Phys. Rev. X* **8** 031086
[29] Shen H, Liu J and Fu L 2018 *Phys. Rev. B* **97** 205140
[30] Torlai G and Melko R G 2017 *Phys. Rev. Lett.* **119** 030501
[31] Fösel T, Tighineanu P, Weiss T and Marquardt F 2018 *Phys. Rev. X* **8** 031084
[32] Baireuther P, O'Brien T E, Tarasinski B and Beenakker C 2018 *Quantum* **2** 48
[33] Andreasson P, Johansson J, Liljestrand S and Granath M 2019 *Quantum* **3** 183
[34] Yuan L, Yang X S and Wang B Z 2019 *Acta Phys. Sin.* **68** 170503 (in Chinese)
[35] Zhu Z J, Guo Y, Yang F, Xiao B J and Li J G 2019 *Chin. Phys. B* **28** 125204
[36] Li P C, Wang H Y, Dai Q and Xiao H 2012 *Acta Phys. Sin.* **61** 160303 (in Chinese)
[37] Han J, Zhang Z, Zhang X and Zhou J 2020 *Chin. Phys. B* **29** 110201
[38] Luchnikov I A, Vintskevich S V, Grigoriev D A and Filippov S N 2020 *Phys. Rev. Lett.* **124** 140502
[39] Perich M G, Gallego J A and Miller L E 2018 *Neuron* **100** 964
[40] Klos C, Kossio Y F K, Goedeke S, Gilra A and Memmesheimer R M 2020 *Phys. Rev. Lett.* **125** 088103
[41] Sussillo D and Abbott L F 2009 *Neuron* **63** 544