

## Imaginary components of out-of-time-order correlator and information scrambling for navigating the learning landscape of a quantum machine learning model


Manas Sajjan <sup>1,4,\*</sup>, Vinit Singh <sup>1,4,\*</sup>, Raja Selvarajan,<sup>2,4</sup> and Sabre Kais <sup>1,2,3,4,†</sup>

<sup>1</sup>*Department of Chemistry, Purdue University, West Lafayette, Indiana 47907, USA*

<sup>2</sup>*Department of Physics and Astronomy, Purdue University, West Lafayette, Indiana 47907, USA*

<sup>3</sup>*Department of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana 47907, USA*

<sup>4</sup>*Purdue Quantum Science and Engineering Institute, Purdue University, West Lafayette, Indiana 47907, USA*

 (Received 30 August 2022; revised 13 January 2023; accepted 23 January 2023; published 27 February 2023)

We introduce and analytically illustrate that hitherto unexplored imaginary components of out-of-time order correlators can provide unprecedented insight into the information scrambling capacity of a graph neural network. Furthermore, we demonstrate that it can be related to conventional measures of correlation like quantum mutual information and rigorously establish the inherent mathematical bounds (both upper and lower bound) jointly shared by such seemingly disparate quantities. To consolidate the geometrical ramifications of such bounds during the dynamical evolution of training we thereafter construct an emergent convex space. This newly designed space offers much surprising information including the saturation of lower bound by the trained network even for physical systems of large sizes, transference, and quantitative mirroring of spin correlation from the simulated physical system across phase boundaries as desirable features within the latent subunits of the network (even though the latent units are directly oblivious to the simulated physical system) and the ability of the network to distinguish exotic spin connectivity (volume law vs area law). Such an analysis demystifies the training of quantum machine learning models by unraveling how quantum information is scrambled through such a network introducing correlation surreptitiously among its constituent subsystems and open a window into the underlying physical mechanism behind the emulative ability of the model.

DOI: [10.1103/PhysRevResearch.5.013146](https://doi.org/10.1103/PhysRevResearch.5.013146)

### I. INTRODUCTION

Heralding machine learning algorithms to be the most disruptive technological advancement of the present era would not be an overstatement [1–4]. Despite successful inroads of the former to enable scientific applications on both classical and quantum hardware [5–11], a pervasive reluctance prevails in making such algorithms mainstream as indicated by a recent survey [12]. A part of the culpability is in the very nature of training of the associated paradigmatic models, which often seems agnostic to physical principles or human-acquired domain intuition. Attempting to address this lacuna, the primary objective of our thesis is to gain physical insight into the learning mechanism of a machine learning model (to be called the learner) assigned to simulate the eigenstates of any user-defined system (to be called the driver), a task central to the core of many physicochemical applications [13].

The major contributions of this paper are manifold. Following a description of the learner, we explicate the role of

the information transport and scrambling between the internal subunits of the learner during the course of its training. To this end, the hitherto unexplored imaginary component of out-of-time order correlators (OTOCs) [14] of the learner is defined and analytically characterized using invariants of motion generated from the underlying Lie algebra [15]. It is then subsequently employed to act as a compass in navigating the parameter landscape during learning. In recent years OTOCs has been used as a quintessential measure of how fast information propagation away from the source of initiation happens in the real-time post any local excitation in atomic systems [16–19], in statistical physics to probe thermalization [20–22], in quantum-information theory [23,24], as a diagnostic tool for quantum chaos [25], and even in models mimicking aspects of quantum gravity [26–29]. Such correlators have also been measured using quantum circuits [30–34]. We thereafter connect such a quantifier with known measures of quantum correlation and illustrate analytically the relative bounds shared by the two quantities, which are stricter than conventionally known bounds. Equipped with these aforesaid probes, we provide a map of navigating the parameter landscape during training of the network in an emergent space and demonstrate with with appropriate case studies features of the trained learner like saturation of lower bound in the above inter-relationship and how the footprints of correlation in the driver get imprinted onto a trained learner, thereby empowering the latter to be used as a concrete diagnostic tool in investigating physical phenomena like phase

\*These authors contributed equally to this work.

†kais@purdue.edu

*Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.*

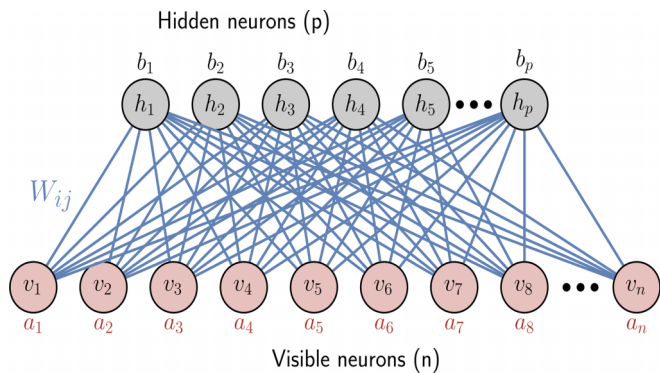


FIG. 1. The description of the learner network  $G = (V, E)$  as defined in text (also known as restricted Boltzmann machine (RBM) in literature [35,36].) The set of hidden neurons  $\{h_j\}_{j=1}^p$  are shown in grey with the corresponding bias vector ( $\vec{b}$ ). Similarly, set of visible neurons  $\{v_i\}_{i=1}^n$  are shown in red with the corresponding bias vector ( $\vec{a}$ ). The interconnecting weight matrix with elements  $W_{ij}$  is shown in blue. The parameter set  $\vec{X} = (\vec{a}, \vec{b}, \vec{W})$  are tuned during the training of the network.

transitions as well as in differentiating between drivers with exotic connectivity/interactions, etc. by simply accessing properties of the learner alone.

In the following section (Sec. II) we shall describe the generative neural network we use for this paper. In Sec. II B we define OTOCs in a general setting. In Sec. II C we introduce and prove that several invariants of motion associated with the phase-space description of OTOC for the aforesaid neural network exists with particular emphasis on the hitherto unexplored imaginary part, which as we shall see which form a key player in our analysis. In Sec. II D we describe the generators associated with the said invariants, which highlights an underlying Lie algebra. In Sec. II E we prove how the imaginary part of OTOC for our network is related to conventional measures of correlation as described previously including the relative bounds, which they share and construct a new emergent convex space to understand the training mechanism of the network and the role of its latent sub-units. In Sec. III we describe the polynomially scaling algorithm for training the network and subsequent construction of the new space. In Sec. IV we describe our primary inferences from numerical studies in the said space and conclude in Sec. V.

## II. THEORETICAL BACKGROUND

### A. Description of the graph neural network $G = (V, E)$

The specific description of the generative network used as the learner in this work is illustrated in Fig. 1. Formally the learner is a connected bipartite graph  $G = (V, E)$  (also known as restricted Boltzmann machine, RBM [35–37]). The set  $V$  consists of  $(p + n)$  neurons with  $(p, n) \in \mathbb{Z}_+$  and is further classified into two subsets as  $V = \{v_i\}_{i=1}^n \cup \{h_j\}_{j=1}^p$  (see Fig. 1). Both the subsets are endowed with a locally accessible  $\sigma^z$  (Pauli-z) operator and their corresponding *bias*-vectors are  $\vec{a} \in \mathcal{R}^n$  and  $\vec{b} \in \mathcal{R}^p$ . The edge set  $E$  can be characterized with the adjacency matrix  $Adj(G) \in \{0, 1\}^{(n+p) \times (n+p)}$  of the graph

$G$  defined as follows:

$$Adj(G)_{ij} = \begin{cases} 1, & \text{if } v_i \in \{v_i\}_{i=1}^n, \quad v_j \in \{h_j\}_{j=1}^p \\ & \forall (v_i, v_j) \in V \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$\forall (i, j) \in \mathcal{Z}^{p+n}$ . Corresponding to each nonzero entry in  $Adj(G)$  we define an edge  $e_{ij} \in E$ . This would mean that  $|E| = p * n$ . Associated with  $e_{ij} \in E$  we define a *weight* matrix  $\vec{W} \in \mathcal{R}^{n \times p}$  (shown in blue in Fig. 1) each element of which quantifies the strength of the shared connection between any of the neurons from the subset  $\{v_i\}_{i=1}^n$  (visible-node register) to every neuron in the subset  $\{h_j\}_{j=1}^p$  (hidden-node register). Collectively the tunable parameters  $\vec{X} = (\vec{a}, \vec{b}, \vec{W}) \in \mathcal{R}^{n+p+np}$  enables us to define the learner's Hamiltonian  $\mathcal{H} : G \mapsto \mathcal{R}^{2^{p+n} \times 2^{p+n}}$  similar to that of a classical Ising model [38,39] as

$$\mathcal{H}(\vec{X}, \vec{v}, \vec{h}) = \sum_{i=1}^n a_i \sigma^z(v_i) + \sum_{j=1}^p b_j \sigma^z(h_j) + \sum_{i=1, j=1}^{n, p} W_j^i \sigma^z(v_i) \sigma^z(h_j), \quad (2)$$

where  $\sigma^z(\chi_i)$  represents operator  $\sigma^z$  acting at neuron  $\chi_i$ . The learner is trained to encode a probability distribution that corresponds to the diagonal elements of a thermal state  $\rho_{th}$  of the Hamiltonian in Eq. (2), and is defined as follows [36,37,40]:

$$\rho_{th}(\vec{X}, \vec{v}, \vec{h}) = \frac{e^{-\mathcal{H}(\vec{X}, \vec{v}, \vec{h})}}{\text{Tr}_{\{v, h\}} e^{-\mathcal{H}(\vec{X}, \vec{v}, \vec{h})}}. \quad (3)$$

Any instance of spin configuration  $(\vec{v}, \vec{h})$  of the combined registers of  $(p + n)$  spins are samples drawn from the said thermal distribution in Eq. (3). Equipped with this, the primary objective of the learner network  $G$  is to mimic the amplitude field of the target state  $\psi(\vec{X})$  of the driver Hamiltonian  $H \in \mathcal{C}^{d \times d}$  following the prescription

$$\psi(\vec{v}, \vec{X}) = \sum_{\vec{h}} \text{diag}(\rho_{th}(\vec{X}, \vec{v}, \vec{h})) \frac{e^{-\sum_{i=1}^n a_i \sigma^z(v_i)} \times \prod_{j=1}^p 2 \cosh(b_j + \sum_i W_j^i \sigma^z(v_i))}{\text{Tr}_{\{v, h\}} e^{-\mathcal{H}(\vec{X}, \vec{v}, \vec{h})}}. \quad (4)$$

Whether the training happens on a classical processor or a NISQ device, the flow of the algorithm, in either case, involves randomly initializing  $\vec{X}$  to construct  $\psi(\vec{v}, \vec{X})$  as in Eq. (4) and then evaluating an appropriate merit function  $J(\vec{X}) = \langle \hat{J} \rangle$  (usually  $\hat{J} = H$  but other properties can be optimized too). The variational parameters  $X = (\vec{a}, \vec{b}, \vec{W})$  are thereafter subsequently tuned using gradient-based updates of the merit function  $\partial_{\vec{X}} J(\vec{X})$  until a desired convergence threshold is reached. The algorithmic details of such a training process can be found elsewhere [41–44]. Formally such a training exploits the isomorphism of the vector space  $\mathcal{C}^d$  of the driver and the space of spin configurations of the visible node  $\vec{v}$  of the learner of dimension  $S = \dim(2^n)$  with  $n = \lceil \log_2 d \rceil$ . As  $\vec{X} \in \mathcal{R}^{n+p+np}$ , we specifically focus on drivers with non-negative coefficients for the target state. Extension to account

for the phase of the target wavefunction is straightforward [45,46].

The number of neurons  $p$  in  $\{h_j\}_{j=1}^p$  is chosen arbitrarily by the user (usually  $p \sim n$ ). It is clear from Eq. (4) that the variational form of the ansatz is independent of  $\sigma^z(h_j)$ , i.e., the spins of the latent neurons and it is the configurations of  $\{\sigma^z(v_i)\}_{i=1}^n$ , which forms the requisite basis for the eigenstate of the driver. While from the optimization point of view, the role of the hidden set of neurons ( $\{h\}_{j=1}^p$ ) is thus to enhance the expressive capability of the network by increasing the number of tunable parameters ( $\vec{v}, \vec{W}$ ), from a more physical perspective  $\{h\}_{j=1}^p$  induces higher-order correlation between the neurons of the visible layer by relaying the information between a given  $(v_i, v_k) \in \vec{v}$  as the latter is devoid of any direct interaction. This relay is sensitive to the connections ( $e_{km} \in E$ ), which defines the *weight* matrix elements  $W_{km}$  of the  $k$ th neuron in  $\vec{v}$  and  $m$ th neuron in  $\vec{h}$ . Central goal of our paper is thus probing how  $\{h_j\}_{j=1}^p$  physically fosters correlation between configurations of  $\{v_i\}_{i=1}^n$  by analyzing how an initial excitation on a given visible neuron is shared with a given hidden neuron in real-time. This will be a direct neuron-resolved picture of the dynamical evolution of the network during training and provide valuable insight into its learning mechanism.

### B. Out of time order correlators (OTOC)

To attain the aforesaid objective of probing information exchange between the active (visible) and latent (hidden) units of the learner, we shall employ an OTOC, which we shall define in detail in this section. An OTOC is primarily composed of two unitary operators  $U_1(0)$  and  $U_2(0)$  wherein  $U_i(0)$  is a local operator for a specific site  $i$  evolving in time  $t$  through Heisenberg prescription [47]. The quantity is sensitive to the extent of information scrambling between sites  $\{1, 2\}$  [48–51] and is defined as follows:

$$C_{U_1, U_2}(t) = \langle U_1^\dagger(0) U_2^\dagger(t) U_1(0) U_2(t) \rangle. \quad (5)$$

Equation (5) is an estimator of growth of the operator  $U_2(t) = e^{iH_{otoc}t} U_2(0) e^{-iH_{otoc}t}$  under the effect of the generator  $H_{otoc}$  assuming the latter possesses interaction within the different sites of the system. If the sites (1,2) are far apart, the supporting bases of the operators  $U_1(0)$  and  $U_2(0)$  are sparsely overlapping and hence  $C_{U_1, U_2}(0)$  will be initially 1. With time, the operator  $U_2(t) = e^{iH_{otoc}t} U_2(0) e^{-iH_{otoc}t}$  will start to extend its support thereby culminating in an eventual overlap between the probe operator  $U_1(0)$  and  $U_2(t)$ , which ultimately lead to changing values of  $C_{U_1, U_2}(t)$ . The quantity  $C_{U_1, U_2}(t)$  thus directly hints at how fast the excitation has traveled from the initially localized point at 2 to site 1. The reason for the nomenclature of “out-of-time order correlator” is due to the fact that the expression  $C_{U_1, U_2}(t)$  has a time ordering, which is nonmonotonic as opposed to forward time correlators like  $\langle U_2(t) U_1(0) \rangle$  wherein the operators  $U_i$  are sequentially arranged in ascending order of time. Also, unlike two-point correlators, which are known to decay in  $O(1)$  time irrespective of the length ( $L$ ) of the system employed, OTOCs like Eq. (5) decay in time proportional to the difference in location of the two sites (hence  $\sim L$ ) [52].

### C. Imaginary component of OTOC of $G = (V, E)$ —Geometrical characterization in phase space

Unlike in most reports wherein the real part of Eq. (5) is used, we introduce the imaginary part of Eq. (5), i.e.,  $\text{Im}(C_{U_1, U_2}(t))$  and shall see that it is also an important player in our analysis. To this end we offer in Appendix A a general formulation for obtaining both the  $\text{Re}(C_{U_1, U_2}(t))$  and  $\text{Im}(C_{U_1, U_2}(t))$  through positive semidefinite construction of other appropriate operators. To understand how  $\{h_j\}_{j=1}^p$  and  $\{v_i\}_{i=1}^n$  scrambles information internally we now construct a specific OTOC and establish the contents of the following theorem.

*Theorem II.1.* For a given parameter vector  $\vec{X}$ , one can define  $\mathcal{H}(\vec{X}, \vec{v}, \vec{h})$  [see Eq. (2)] and a thermal state  $\rho_{th}(\vec{X}, \vec{v}, \vec{h})$ . Let us thereafter define the following OTOC with  $U_1(0) = \tilde{\sigma}_\alpha = \sigma^\alpha(v_k, 0) - \kappa_1 \mathcal{I}$ , and operator  $U_2(0) = \tilde{\sigma}_\beta = \sigma^\beta(h_m, 0) - \kappa_2 \mathcal{I}$  and the generator  $H_{otoc} = \mathcal{H}(\vec{X}, \vec{v}, \vec{h})$  in Eq. (5)  $\forall \{\alpha, \beta\} \in \{x, y\}$ .

$$C_{\sigma^\alpha, \sigma^\beta}(\kappa_1, \kappa_2, \vec{X}, t) = \langle \tilde{\sigma}^\alpha(v_k, 0) \tilde{\sigma}^\beta(h_m, t) \tilde{\sigma}^\alpha(v_k, 0) \tilde{\sigma}^\beta(h_m, t) \rangle \quad (6)$$

Note that  $\{\kappa_1, \kappa_2\} \in \mathcal{C}^2$  are arbitrary user-defined mean translations. Also  $\langle \cdot \rangle$  indicates averaging over the thermal state  $\rho_{th}(\vec{X}, \vec{v}, \vec{h})$  defined in Eq. (3), which activates the  $\vec{X}$  dependence. Using 6, one can then make the following statements:

(1) For  $(\kappa_1, \kappa_2) \in \mathcal{C}^2$

$$C_{\sigma^\alpha, \sigma^\beta}(\kappa_1, \kappa_2, \vec{X}, t) = C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, t) + |\kappa_1|^2 |\kappa_2|^2 + |\kappa_2|^2 + |\kappa_1|^2$$

(2) The following invariants of motion exists for  $C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, t)$ :

$$(a) I_1 = -2\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) \text{Cos}(\tau) - 2\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) \text{Sin}(\tau)$$

$$(b) I_2 = -2\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) \text{Sin}(\tau) + 2\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) \text{Cos}(\tau)$$

where  $\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau)$  can either be the real or the imaginary part

of  $(C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau))$  and  $\square$  is  $\frac{\partial}{\partial \tau}$  with  $\tau = 4W_m^k t$

*Proof.* See Appendix C. ■

A combination of the invariants from Theorem II.1(2) is plotted in Figs. 2(a) and 2(b) for the  $\text{Re}(C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, t))$  and  $\text{Im}(C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, t))$  and shows decisively the importance of the imaginary part. Similar results for other invariants are illustrated in Appendix C. Certain assertions are apparent from Theorem II.1. Firstly, as a direct corollary [proven in Appendix C as Corollary Eq. (5.1)], one can deduce analytical expressions for  $C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)$  as

$$C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) = \cos(\tau) + i \langle \sigma^z(v_k, 0) \sigma^z(h_m, 0) \rangle_{\rho_{th}(\vec{X})} \sin(\tau) \quad (7)$$

Equation (7) guarantees that temporal behavior of the OTOC in Eq. (6) is oscillatory in nature due to the  $SU(2)$  algebra associated with the unitary rotation of  $\sigma^\beta(h_m, t)$  ( $\beta \in \{x, y\}$ ) around the  $z$  axis induced by the generator  $H_{otoc} = \mathcal{H}$  in Eq. (2). The frequency associated with the rotation for both terms in Eq. (7) is expectedly dictated by the strength of the interaction ( $W_m^k$ ) shared by the  $k$ th visible neuron and  $m$ th hidden neuron. Note that training of  $G$  amounts to hopping between OTOC trajectories (each of which is generated with a frozen incumbent instance of  $\vec{X}$ ) in phase space as we traverse

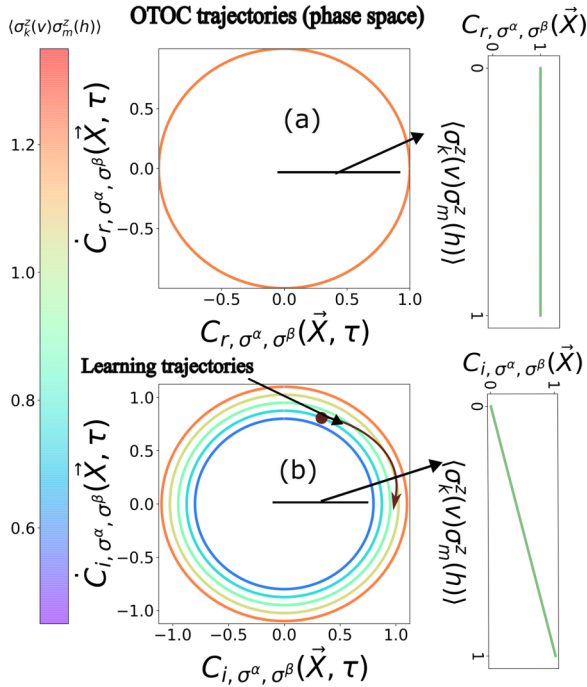


FIG. 2. (a) The profile for the real part of the compound invariant  $\frac{I_1^2 + I_2^2}{4}$  (see Appendix C, Corollary 5.2) obtained from the simple invariants of Theorem II.1 where  $\alpha = \beta = x$  and  $C_{r,\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau) = \text{Re}(C_{\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau))$  has been substituted for  $\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau)$  in Theorem II.1(2) (b) Similar to (a) but for  $C_{i,\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau) = \text{Im}(C_{\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau))$ . Note the subscript “r” (“i”) denotes the real(imaginary) part of Eq. (6) in the plots. Both quantities are evaluated at  $(\kappa_1 = 0, \kappa_2 = 0)$ , hence the explicit dependence on  $(\kappa_1, \kappa_2)$  has been dropped for notational brevity. The trajectory in the phase space for both cases is a circle (Corollary 5.2) whose radius remains conserved during training in (a) but changes in (b). This is further corroborated by the two green subplots alongside, which for (b) shows that the radius is exactly equal to  $\langle \sigma^x(v_k, 0), \sigma^x(h_m, 0) \rangle_{\rho_{ih}(\vec{X}, \vec{v}, \vec{h})}$  and is sensitive to  $\vec{X}$  unlike in (a). This indicates why the  $\text{Im}(C_{\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau))$  of the OTOC string in Eq. (6) can yield important insight about the training process of  $G$ .

the parameter space by varying  $\vec{X}$  [see Fig. 2(b)]. We see that the amplitude of the  $\text{Im}(C_{\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, t))$  yields directly a two-body correlation function  $\langle \sigma^z(v_k, 0), \sigma^z(h_m, 0) \rangle_{\rho_{ih}(\vec{X}, \vec{v}, \vec{h})}$  of spins in the visible and hidden register of the learner [see Fig. 2(b)], which is sensitive to  $\vec{X}$  and thus probes the changing correlation content between  $h_m$  and  $v_k$  during training.

### D. Lie-algebraic generators associated with invariants of OTOC

It is possible to define generators associated with the invariants defined in Theorem II.1 possessing an underlying Lie-algebraic structure. For example for the invariant type given in Theorem II.1(1) with the following expression:

$$I_1 = -2\dot{\xi}_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau)\cos(\tau) - 2\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau)\sin(\tau), \quad (8)$$

where  $\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau)$  can be either  $C_{r,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)$  or  $C_{i,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)$  one can deduce the following transforma-

tion  $\hat{A}(\phi, \tau)$ :

$$\begin{aligned} \hat{A}(\phi, \tau)(\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau)) &= e^{\phi \cos(\tau) \frac{\partial}{\partial \xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau)}} (\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau)) \\ &= \xi'_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau). \end{aligned} \quad (9)$$

Note that the derivative in the exponent is with respect to  $\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau)$  itself. Such a transformation  $\hat{A}(\phi, \tau) : \{\xi\}_{I_1} \mapsto \{\xi'\}_{I_1}$  where  $\{\xi\}_{I_1}$  is a solution space marked by the given value of the invariant  $I_1$ . In other words, a given solution  $\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau)$  with a specific value of the invariant  $I_1$ , the transformation changes it to another solution  $\xi'_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau)$ , which possesses the same value for the invariant. This can be verified by explicit computation too. For instance, if  $\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau) = C_{r,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)$ , then from the Corollary 5.1 it is clear that  $\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau) = C_{r,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau) = \cos(W_m^k t) = \cos(\tau)$ . Substituting this in Eq. (8), one gets the value of  $I_1$  as 0. If the transformation  $A(\phi, \tau)$  is applied on  $\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau) = \cos(\tau)$ , the new solution is  $(1 + \phi)\cos(\tau)$ , which has the same value of the invariant as before. Similarly, one can also deduce an invariant-preserving transformation for  $I_2$  type invariants in

Theorem II.1(2) as  $e^{\phi \sin(\tau) \frac{\partial}{\partial \xi_{\sigma^\alpha,\sigma^\beta}}}$ . It can be shown that the generators of the two transformations commutes. For other invariants in Corollary 5.2 one can similarly deduce other transformations like this. The generators of a full set of such transformations forms a closed single parameter Lie group (with respect to parameter  $\phi$ ), which can be shown using their commutation algebra. Such a structure is characteristic of systems with harmonic degrees of freedom, but discovering it within the abstract phase space of OTOC string for the learner is interesting and worth further investigation. The ramifications of such generators on the full phase-space of the OTOC strings and its effect on the training of the learner will be explored in the future. We thus see that instead of a direct evaluation of the OTOC, evaluation through the invariants described in Theorem 1 offers a fuller characterization of the phase space with richer insight into the geometry of the manifold.

### E. Inter-relationship with covariance and quantum mutual information ( $\mathcal{I}(v_k, h_m)$ )—Mapping training trajectories to $\mathcal{I} - \eta(\vec{X})$ space

We further consolidate the importance of  $\text{Im}(C_{\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, t))$  concretely in this section by showing that the quantity can be related to quantum mutual information. To do that it is imperative to first establish a direct relationship of the said quantity with  $\text{Cov}(\sigma^z(v_k, 0), \sigma^z(h_m, 0))_{\rho_{ih}}$  by using the results of Theorem II.1(1–2) as follows,

$$\begin{aligned} \eta(\vec{X}) &= \text{Cov}(\sigma^z(v_k, 0), \sigma^z(h_m, 0))_{\rho_{ih}} \\ &= C_{\sigma^\alpha,\sigma^\beta}(\kappa_1, 0, \vec{X}, t_1) + C_{\sigma^\alpha,\sigma^\beta}(0, \kappa_2, \vec{X}, t_1) \\ &\quad - C_{\sigma^\alpha,\sigma^\beta}(\kappa_1, \kappa_2, \vec{X}, t_1) \end{aligned} \quad (10)$$

with  $\kappa_1 = \sqrt{i\langle \sigma^z(v_k, 0) \rangle}$ ,  $\kappa_2 = \sqrt{i\langle \sigma^z(h_m, 0) \rangle}$  and  $t_1 = \frac{\pi}{8W_m^k}$ .  $\eta(\vec{X})$  in Eq. (10) can only attain a value of zero if the neurons  $v_k$  and  $h_m$  are uncorrelated. We shall now connect  $\eta(\vec{X})$  to other well-known correlation measures like Von-Neumann

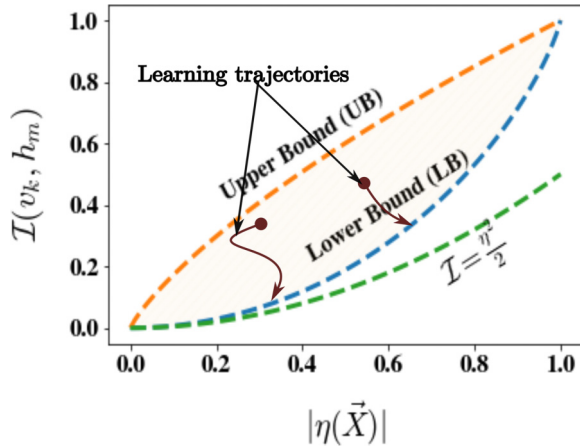


FIG. 3. The upper bound(UB) and lower bound (LB) in the convex  $\mathcal{I} - \eta$  space for network  $G$  as described in Theorem II.2 in Eqs. (12) and (13) respectively. In all plots in this panel, we shall use  $\alpha = \beta = x$  in Eq. (10). Provided alongside in green is the conventionally known lower bound for any general bipartition of an arbitrary system [57]. The lower bound (LB) for  $G$  is thus stricter than the known general bound. Representative learning trajectories of the network  $G$  are shown.

entropies of and incipient mutual information ( $\mathcal{I}(v_k, h_m)$ ) [53–56] of subsystems of  $G$  using the following theorem.

**Theorem II.2.** The two particle-density matrices  ${}^2\rho(v_k, h_m)$  and one particle-density matrices  ${}^1\rho(v_k)$  and  ${}^1\rho(h_m)$  of the learner  $G$  can be computed for a given  $\vec{X}$  from Eq. (3) (see Appendix E). Using these one can construct  $\mathcal{I}(v_k, h_m) = S({}^1\rho(v_k)) + S({}^1\rho(h_m)) - S({}^2\rho(v_k, h_m))$  where  $S(Y) = -Tr(Y \ln Y)$ . The following statement is true

$$LB \leq \mathcal{I}(v_k, h_m) \leq UB \quad (11)$$

wherein the lower bound (LB) is

$$LB = 2 - 2\mathcal{L}\left(\frac{1 + \eta(\vec{X})}{4}\right) - 2\mathcal{L}\left(\frac{1 - \eta(\vec{X})}{4}\right) \quad (12)$$

and the corresponding upper bound (UB) is

$$UB = \mathcal{L}\left(\frac{1 + \sqrt{1 + (-1)^\gamma \eta(\vec{X})}}{2}\right) + \mathcal{L}\left(\frac{1 - \sqrt{1 + (-1)^\gamma \eta(\vec{X})}}{2}\right) \quad (13)$$

with  $-1 \leq \eta(\vec{X}) \leq 1$  as defined in Eq. (10) and  $\mathcal{L}(x) = -x \ln(x) \forall x \in \mathcal{R}_+$  and  $\gamma = 0$  if  $\eta(\vec{X}) < 0$  or  $\gamma = 1$  if  $\eta(\vec{X}) \geq 0$

*Proof.* See Appendix E. ■

The implication of Theorem II.2 is profound and is illustrated diagrammatically in Fig. 3. It introduces a newly emergent  $\mathcal{I} - \eta$  space to probe learning trajectories. In this space, the bounds together define a convex set (see Fig. 3) within which resides the acceptable values of  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$ , that the learner  $G$  can access during the course of its training. One must note that LB in Eq. (12) is more stringent

compared to previously known bound of  $\frac{\eta(\vec{X})^2}{2}$  [57] as seen in Fig. 3.

### III. COMPUTATIONAL DETAILS

#### A. Training algorithm and efficient polynomially scaling construction of $\mathcal{I} - \eta(\vec{X})$ space

In this section we discuss in detail the computational algorithm used for training the network  $G = (V, E)$  and for subsequent estimation of  $\mathcal{I}$  and  $\eta(\vec{X})$  through sampling to extract the features of the learner in this emergent space.

For a given driver Hamiltonian  $H \in \mathcal{C}^{d \times d}$ , the variational form of the target state, which is used to train the learner as ansatz is defined in Eq. (4). Since we are interested in ground state of the respective drivers, training as explained before is done minimizing the cost function  $J(\vec{X})$  (in this case the energy) with respect to the parameters  $\vec{X}$  of the trial ansatz  $\psi(\vec{v}, \vec{X})$  as follows:

$$J(\vec{X}) = \frac{\langle \psi(\vec{v}, \vec{X}) | H | \psi(\vec{v}, \vec{X}) \rangle}{\langle \psi(\vec{v}, \vec{X}) | \psi(\vec{v}, \vec{X}) \rangle}. \quad (14)$$

For updating the parameters of Eq. (14) we follow the algorithm in Ref. [58] closely, which is based on Sorella's stochastic reconfiguration technique [59]. In this technique the parameters of the cost function are updated using a preconditioner  $F \in \mathcal{C}^{p*n \times p*n}$  defined as follows:

$$\vec{X} \rightarrow \vec{X} - l * F^{-1}(\vec{X}) S(\vec{X}), \quad (15)$$

where

$$F_j^i = \langle D_i^\dagger D_j \rangle - \langle D_i^\dagger \rangle \langle D_j \rangle \quad (16)$$

$$S_i = \langle E_{\text{loc}} D_i^\dagger \rangle - \langle E_{\text{loc}} \rangle \langle D_i^\dagger \rangle \quad (17)$$

$$D_i |\vec{v}\rangle = \frac{\partial_{\vec{X}_i} \psi(\vec{v}, \vec{X})}{\psi(\vec{v}, \vec{X})} |\vec{v}\rangle \quad \forall \vec{X}_i \in (\vec{a}, \vec{b}, \vec{W}) \quad (18)$$

$$E_{\text{loc}} = \frac{\langle \vec{v} | H | \psi(\vec{v}, \vec{X}) \rangle}{\psi(\vec{v}, \vec{X})}. \quad (19)$$

The indices ( $i, j$ ) in Eqs. (15)–(19) run over all  $p * n$  parameters in  $\vec{X}$ . Also  $l \in \mathcal{R}_+$  is known as the learning rate and  $|\vec{v}\rangle$  is the computational basis of configuration of the visible node register. The above expressions are true for any variational ansatz. For the particular graph neural network encoding  $G = (V, E)$ , the components of  $D_i$  can be analytically expressed as diagonal matrix elements in the computational basis as follows:

$$\langle \vec{v}' | D_{a_i} | \vec{v} \rangle = \sigma^z(v_i) \delta_{\vec{v}' \vec{v}}, \quad (20)$$

$$\langle \vec{v}' | D_{b_j} | \vec{v} \rangle = \tanh\left(b_j + \sum_i W_j^i \sigma^z(v_i)\right) \delta_{\vec{v}' \vec{v}}, \quad (21)$$

$$\langle \vec{v}' | D_{W_j^i} | \vec{v} \rangle = \sigma^z(v_i) \tanh\left(b_j + \sum_i W_j^i \sigma^z(v_i)\right) \delta_{\vec{v}' \vec{v}}. \quad (22)$$

The averaging  $\langle \dots \rangle$  defined in Eqs. (16) and (17) is over the distribution  $\frac{|\psi(\vec{v}, \vec{X})|^2}{\langle \psi(\vec{v}, \vec{X}) | \psi(\vec{v}, \vec{X}) \rangle}$ . This distribution in accordance to Eq. (4) is dependant on configurations of  $\{\sigma(v_i)\}_{i=1}^n$ . Exactly

computing all such  $2^n$  configurations would yield an exponentially scaling protocol. Hence the algorithm in Ref. [58] uses a Monte Carlo based sampling protocol using Metropolis-Hastings procedure. The primary workhorse of the protocol relies on starting with one of the arbitrary [say  $\{\sigma(v_i)\}_{i=1}^n$ ]  $2^n$  configuration and then randomly choosing one of the spins  $\sigma(v_i)$  and mutating/flipping it to generate a new configuration [say  $\{\sigma(v_i)'\}_{i=1}^n$ ]. The new configuration is accepted or rejected using the following rule:

$$P(\{\sigma(v_i)\}_{i=1}^n \rightarrow \{\sigma(v_i)'\}_{i=1}^n) = \min\left(1, \left|\frac{\psi(\vec{v}', \vec{X})}{\psi(\vec{v}, \vec{X})}\right|^2\right). \quad (23)$$

Reference [58] establishes that the computational cost of each such Monte Carlo sweep is  $O(n * p)$ . If the number of sweeps used is  $N_S$  then the total cost of computing the gradient updates by computing  $F$  would be  $O(n * p * N_S)$  as there are  $O(n * p)$  parameters over which the matrix  $F$  and vector  $S$  needs to be computed. We use Netket library [60] to perform computations using this algorithm. If the condition number of  $F$  is large, invertibility may be a problem, which is obviated

using a infinitesimal shift, which has been fixed at 0.01 in our calculations. The total number of independent Markov chains is set to 1000 and 60 sweeps are used at each step along a chain. All parameters of  $\vec{X}$  is randomly initialized from a normal distribution of zero mean and 0.01 standard deviation. The maximum number of iterations kept were 200 and learning rate ( $l$ ) for training  $G = (V, E)$  is set to 0.1. With these set of parameters we were able to achieve an error threshold of  $\leq 0.1\%$  for convergence.

Once the training of  $G = (V, E)$  has commenced to the desired accuracy threshold, we have now obtained the trained parameter vector [say  $\vec{X}^* = (\vec{a}^*, \vec{b}^*, \vec{W}^*)$  where  $*$  is not complex conjugation as all parameters are real but denotes a specific instance of the trained  $\vec{X}$  procured after training] of the learner  $G$ . Using these one can easily construct the eigenvalues of the two  ${}^2\rho(v_k, h_m)$  and one-particle reduced density matrices [ ${}^1\rho(v_k)$  or  ${}^1\rho(h_m)$ ] as obtained from the contraction of Eq. (3). The four eigenvalues  $\{\lambda_i({}^2\rho(v_k, h_m))\}_{i=1}^4$  of the two-particle density matrix  ${}^2\rho(v_k, h_m)$  for the learner  $G$  between a specific pair of visible and hidden spins [say  $(k, m)$ ] as deduced in Appendix D 1 are expressed as follows:

$$\lambda_1({}^2\rho(v_k, h_m)) = \lambda({}^2\rho(v_k = 1, h_m = 1)) = \frac{N_1}{Z} e^{-a_k^* - b_m^* - W_m^{k*}} \times \left\langle \prod_{j \neq m}^p 2 \cosh\left(b_j^* + \sum_{i \neq k}^n W_j^{i*} v_i + W_j^{k*}\right) \right\rangle_{P(\{v_i\}_{i \neq k}^n, h_m = 1)} \quad (24)$$

$$\lambda_2({}^2\rho(v_k, h_m)) = \lambda({}^2\rho(v_k = 1, h_m = -1)) = \frac{N_{-1}}{Z} e^{-a_k^* + b_m^* + W_m^{k*}} \times \left\langle \prod_{j \neq m}^p 2 \cosh\left(b_j^* + \sum_{i \neq k}^n W_j^{i*} v_i + W_j^{k*}\right) \right\rangle_{P(\{v_i\}_{i \neq k}^n, h_m = -1)} \quad (25)$$

$$\lambda_3({}^2\rho(v_k, h_m)) = \lambda({}^2\rho(v_k = -1, h_m = 1)) = \frac{N_1}{Z} e^{a_k^* - b_m^* + W_m^{k*}} \times \left\langle \prod_{j \neq m}^p 2 \cosh\left(b_j^* + \sum_{i \neq k}^n W_j^{i*} v_i - W_j^{k*}\right) \right\rangle_{P(\{v_i\}_{i \neq k}^n, h_m = 1)} \quad (26)$$

$$\lambda_4({}^2\rho(v_k, h_m)) = \lambda({}^2\rho(v_k = -1, h_m = -1)) = \frac{N_{-1}}{Z} e^{a_k^* + b_m^* - W_m^{k*}} \times \left\langle \prod_{j \neq m}^p 2 \cosh\left(b_j^* + \sum_{i \neq k}^n W_j^{i*} v_i - W_j^{k*}\right) \right\rangle_{P(\{v_i\}_{i \neq k}^n, h_m = -1)} \quad (27)$$

where each of the averages in Eqs. (24)–(27) are computed over the distribution  $P(\{v_i\}_{i \neq k}^n, h_m)$  and  $N_{h_m}$  is the associated normalization constant. These are defined as

$$P(\{v_i\}_{i \neq k}^n, h_m) = \frac{e^{-\sum_{i \neq k}^n a_i^* v_i + \sum_{i \neq k}^n W_m^{i*} v_i h_m}}{N_{h_m}}, \quad (28)$$

$$N_{h_m} = \prod_{i \neq k}^n 2 \cosh\left(a_i^* + \sum_{i \neq k}^n W_m^{i*} h_m\right). \quad (29)$$

The corresponding eigenvectors of the two particle density matrix for the eigenvalues in Eqs. (24)–(27) are  $|0(v_k)0(h_m)\rangle$ ,  $|0(v_k)1(h_m)\rangle$ , and  $|1(v_k)0(h_m)\rangle$ ,  $|1(v_k)1(h_m)\rangle$ , respectively for the four eigenvalues Eqs. (24)–(27) where  $(0,1)$  is notationally equivalent to  $(1, -1)$  for each spins  $(v_k, h_m)$ .

The quantity  $Z$  is the partition function defined as

$$Z = \sum_{(\vec{v}, \vec{h})} e^{-\sum_i -a_i^* v_i - \sum_i^m b_i^* h_i - \sum_{i,j}^n W_j^{i*} v_i h_j}. \quad (30)$$

However,  $Z$  need not be explicitly computed as it can be eliminated using the unit normalization condition of the eigenvalues. The corresponding eigenvalues for one-particle

density matrix  ${}^1\rho(\xi_i, 0)$  for a neuron  $\xi_i$  in the learner  $G$ , can thereafter be constructed by contraction from Eqs. (24)–(27) are

$$\begin{aligned} \lambda_1({}^1\rho(\xi_i)) &= \lambda_i({}^2\rho(v_k, h_m)) + \lambda_j({}^2\rho(v_k, h_m)) \\ &\quad (\text{if } \xi_i = v_k, (i, j) = (1, 3)) \\ &\quad (\text{if } \xi_i = h_m, (i, j) = (1, 4)), \end{aligned} \quad (31)$$

$$\begin{aligned} \lambda_2({}^1\rho(\xi_i)) &= \lambda_i({}^2\rho(v_k, h_m)) + \lambda_j({}^2\rho(v_k, h_m)) \\ &\quad (\text{if } \xi_i = v_k, (i, j) = (2, 4)) \\ &\quad \text{if } (\xi_i = h_m, (i, j) = (2, 3)), \end{aligned} \quad (32)$$

with respective eigenvectors are  $|0(\xi_i)\rangle$  and  $|1(\xi_i)\rangle$  where  $\xi_i \in (v_k, h_m)$ .

Using the eigenvalues of  ${}^2\rho(v_k, h_m)$  and one-particle reduced density matrices ( ${}^1\rho(v_k)$  or  ${}^1\rho(h_m)$ ) one can compute  $I(v_k, h_m)$  vs  $|\eta(\vec{X})|$  as illustrated in Appendix E and construct the entire space. From Eqs. (24)–(27) and (29) it is clear that the underlying probability distribution  $P(\{v_i\}_{i=1, i \neq k}^n, h_m)$  from which the eigenvalues are computed is defined over the  $2^{n-1}$

configurations  $\{\sigma(v_i)\}_{i \neq k}^n$  of the visible node register (also independent of  $\{h_i\}_{i \neq m}$  but is dependant on  $h_m = \pm 1$ , which is kept fixed for a given eigenvalue). Accurate estimation of the entire distribution would require exponentially scaling resources. However, we shall now demonstrate a polynomially scaling algorithm based on Gibbs sampling technique. The distribution can be marginalized easily as it is completely factorizable over individual  $\{v_i\}_{i \neq k}$  spins due to lack of connectivity among the visible spins in the network  $G$ , which is the key feature required in our sampling. We estimate the eigenvalues using Gibbs sampling from this distribution (29). For each pair  $(k, m)$  of visible and hidden neurons, the sampling technique for a given drawn/sampled configuration of visible neurons (say  $\bar{v}_1$ ) performs a sum over all visible neurons to construct each of the primitive operators  $2\cosh(b_j^* + \sum_{i \neq k}^n W_j^{i*} v_i + W_j^{k*})$ . Each such primitive operator is indexed by  $j$ . The primitive operators are then constructed for every  $j \in \{1, 2 \dots p\}$  and then multiplied together to yield the compound operator  $\prod_{j \neq m} 2\cosh(b_j^* + \sum_{i \neq k}^n W_j^{i*} v_i + W_j^{k*})$ . This step thus incurs a cost of  $O(np)$  alone. This step is repeated for many drawn samples of visible node configurations (say  $\bar{v}_2$  now) with the total number of samples drawn be  $N_E$  thereby introducing a total cost of  $O(np * N_E)$  for this step. This yields the four eigenvalues as given in Eqs. (24)–(27) and completes the computation of  $I(v_k, h_m)$  vs  $|\eta(\bar{X})|$  for one pair of  $(k, m)$  neurons. The entire process is repeated for every pair of  $(k, m)$  neurons. Since there are  $O(np)$  neurons and each of which gives a two particle density matrix, the total computation of all  $pn$  two-particle density matrices incurs a cost of  $O(n^2 p^2 * N_E)$ . Thus from start to finish our entire protocol of training the network  $G = (V, E)$  and the subsequent construction of the  $I(v_k, h_m)$  vs  $|\eta(\bar{X})|$  space is  $O(\text{poly}(n, p))$ .

### B. Hamiltonian of the drivers

To exemplify the consequences further, we now use two drivers namely the transverse field Ising model (TFIM) and the concentric-TFIM [61] (c-TFIM) for a system of  $N = 4, 6, 8, 10, 12, 14, 16, 18, 20, 24$  spins. The generic Hamiltonian for the drivers can be written as

$$H = -B \sum_{i_d} \sigma^x(i_d) - \sum_{i_d j_d} J_{i_d j_d} \sigma^z(i_d) \sigma^z(j_d). \quad (33)$$

For TFIM the matrix elements of  $J_{i_d j_d}$  are

$$J_{i_d j_d} = \begin{cases} J_0, & \text{if } i_d j_d = i_d \pm 1 \quad \forall i_d \\ 0 & \text{otherwise} \end{cases}. \quad (34)$$

For c-TFIM, the elements  $J_{i_d j_d}$  are

$$J_{i_d j_d} = \begin{cases} J_0, & \text{if } i_d = \frac{N}{2} - (q - 1), \\ & j_d = \frac{N}{2} + q \\ 0 & \text{otherwise} \end{cases} \quad (35)$$

with  $\forall q \in \left[1, \frac{N}{2}\right]$ .

While Eq. (34) due to nearest-neighbor interactions [see Fig. 4(a)] admits an area-law scaling ground state, which can only be augmented to a logarithmic correction [62,63], the connectivity graph of Eq. (35) [see Fig. 4(b)] necessitates a volume-law scaling (refer to Appendix I for direct corroboration). Since we choose  $(B \geq 0, J_0 \geq 0)$ , the ground state of both drivers have non-negative coefficients due to Perron-Frobenius theorem [64,65] and undergoes a phase transition from an ordered ferromagnet to the disordered phase owing to spontaneous breaking of  $\mathbb{Z}_2$  symmetry  $[\sigma^z(i_d) \rightarrow -\sigma^z(i_d)$  or  $\pi$  rotation around  $\sigma^x(i_d)]$  as  $g = \frac{B}{J_0}$  is enhanced.

## IV. RESULTS AND DISCUSSION

To simulate the drivers discussed in Sec. III B for  $N = 4, 6, 8, 10, 12, 14, 16, 18, 20, 24$  spins, we use  $n = p = N$  in the learner  $G$  and run several numerical experiments with different randomly chosen initial parameters with the error threshold for convergence set to  $\leq 0.1\%$  (see Sec. III for the training algorithm and Appendix D for convergence plots of training) for each. To see the effect of changing hidden node density  $p/n$  see Appendix G. We use the finally converged  $\bar{X}^*$  obtained from the training to construct the eigenvalues of  ${}^2\rho(v_k, h_m)$  and one-particle reduced density matrices  $[{}^1\rho(v_k)$  or  ${}^1\rho(h_m)]$  and eventually compute  $I(v_k, h_m)$  and  $|\eta(\bar{X})|$  as illustrated in Sec. III. We do this for each pair  $(k, m)$  choosing one from the set of visible and the other from the set of hidden neurons in the learner  $G = (V, E)$ . We display the results of our computation in the  $\mathcal{I} - \eta$  space (illustrated in Fig. 3) in Figs. 4(c)–4(j) for all such pairs at various  $g$  values of the drivers (see Sec. III B for a definition of  $g$ ). For  $N = 4, 6, 10, 20$  in the respective models, we see surprisingly that the representation chosen by the trained learner in the  $\mathcal{I} - \eta$  space always saturates LB in Figs. 4(c)–4(f) and slides along it monotonically for  $g \in (0, \infty]$ . For  $g = 0$ , the representation of  $G$  is devoid of any correlation between subsystems of the visible and hidden neurons  $\forall(k, m)$  with a cluster of points near the green dot (0,0) [especially marked in Figs. 4(c) and 4(g) but is true for all plots in Figs. 4(c)–4(j)] corresponding to the uncorrelated, twofold degenerate ferromagnetic ground state  $(|0000\rangle$  or  $|1111\rangle)$  [64]. For a direct corroboration for all other sizes and in a wide variety of other driver spin models even beyond the drivers discussed in Sec. III B, see Appendix F. From such numerical evidences, we infer a newly discovered learning principle that has never been discussed or investigated before. For a wide variety of drivers, we see that the trained network  $G = (V, E)$  when entrusted with learning a probability density function to mimic the amplitude field of a desired quantum state, invariably chooses a representation that minimizes mutual information  $[\mathcal{I}(v_k, h_m)]$  between the visible and the hidden sub-units for a given covariance  $[\eta(\bar{X})]$ . This further highlights the importance of the  $\mathcal{I}(v_k, h_m)$  vs  $\eta(\bar{X})$  space we investigate here. The pursuance of universality of this result through a formal proof of the statement or under what conditions it fails if at all may be undertaken in the future.

While lower bound saturation remains true at all values of  $g$ , as is evident from Figs. 4(c) and 4(j) that for  $0 \leq g \leq 1$  the density of points associated with the representation chosen by  $G$  shifts dramatically away from the green dot towards

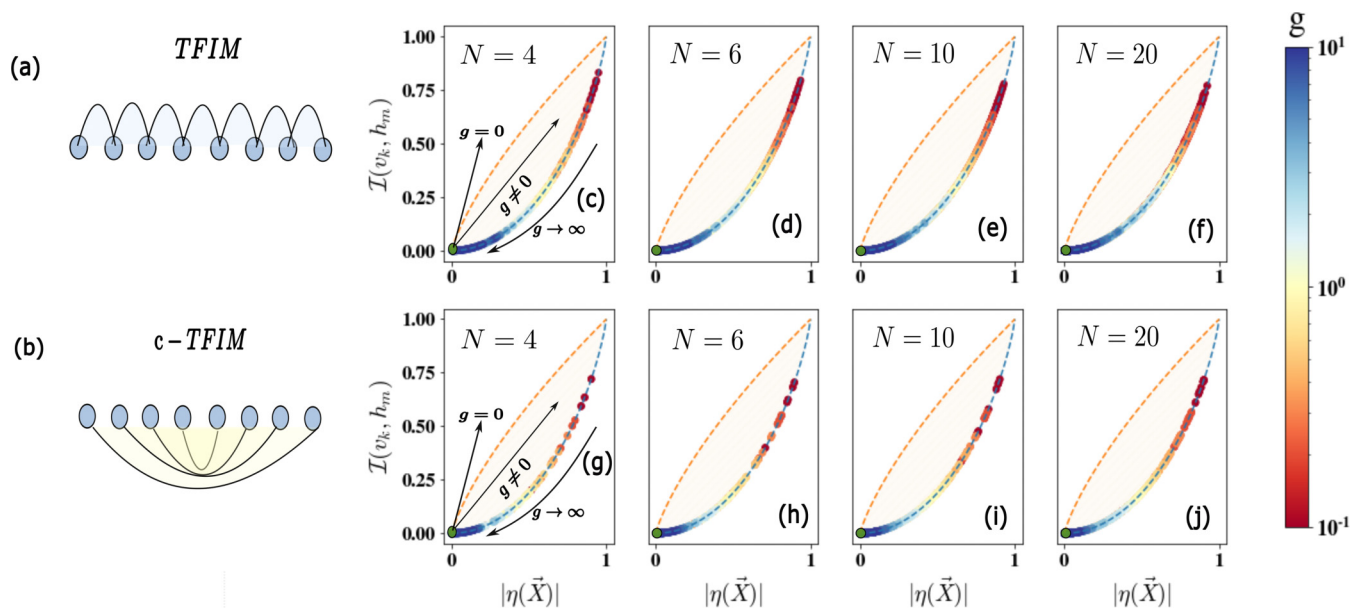


FIG. 4. (a) The schematics of the interactions among the spin registers in the area-law entangled ground state of the driver Hamiltonian defined in Eq. (34) (TFIM) [66–68]. (b) The schematics of the interactions among the spin registers in the volume-law entangled ground state of another driver system defined in Eq. (35) (c-TFIM) [61] (see text for more details and Appendix I). The  $\mathcal{I}(v_k, h_m)$  and  $|\eta(\vec{X})|$  values corresponding to the finally converged parameter  $\vec{X}^*$  of the trained network  $G = (V, E)$  for the ground state of Eq. (34) for (c)  $N = 4$ , (d)  $N = 6$ , (e)  $N = 10$ , and (f)  $N = 20$  spins in the driver in Eq. (34) (TFIM). The  $\mathcal{I}(v_k, h_m)$  and  $|\eta(\vec{X})|$  values corresponding to the finally converged parameter  $\vec{X}$  of the trained network  $G$  for the ground state of Eq. (35) (c-TFIM) for (g)  $N = 4$ , (h)  $N = 6$ , (i)  $N = 10$ , and (j)  $N = 20$  spins in the driver. In all eight plots (c)–(j) we see that the representation chosen by the trained network in the emergent  $\mathcal{I} - \eta$  space always saturates the LB [Eq. (12)] for all  $g \in (0, \infty)$ . The behavior remains intact irrespective of the changing size of the driver system and has been observed for a wider class of drivers too arising in different physical problems (see also Appendix F).

the red points and then eventually returns back towards the uncorrelated state (blue points) [ $\forall (k \in [n], m \in [p])$  where  $n = p = N$ , i.e.,  $n * p$  pairs of  $(k, m)$  values]. This is explicitly marked in Figs. 4(c) and 4(g) but is true for all plots in Figs. 4(c)–4(j). To study the details of such an occurrence and consolidate the observation, we plot in Figs. 5(a)–5(d) the sample means of  $\mathcal{I}(v_k, h_m)$  and  $|\eta(\vec{X})|$  as constructed from the finally converged  $\vec{X}^*$  by training  $G$  averaged not only over all pairs  $(k, m)$  for a given experiment but also over converged runs arising from different initialization (i.e., the averages of points plotted along LB in Figs. 4(c)–4(j) over all pairs of  $(k, m)$  and over several initialization such that for a given  $g$  there is a single representative  $\mathcal{I}(v_k, h_m)$  and  $|\eta(\vec{X})|$  value). To get better idea about the statistic, the standard deviations associated with the averaging process and the standard error of the mean is displayed in Appendix H. For  $g \rightarrow 0_+$  the ground state of the driver exhibits a superposition of several bit-strings/spin configurations, which the network  $G$  now mirrors by choosing a representation that has a significantly higher  $\mathcal{I}(v_k, h_m)$  and  $|\eta(\vec{X})|$  (and hence correlation) among the bi-partitions between visible and hidden neurons [red dots in Figs. 4(c)–4(j)]. This is further reflected in the higher mean  $\mathcal{I}(v_k, h_m)$  in Figs. 5(a) and 5(c) and higher mean  $|\eta(\vec{X})|$  in Figs. 5(b) and 5(d) for each nonzero value of  $g$  than in the  $g = 0$  case.

For  $g \rightarrow \infty$ , the drivers once again display an uncorrelated state with each spin in state  $\frac{|0\rangle + |1\rangle}{\sqrt{2}}$ . To ape this limit, the representation chosen by  $G$  thus gradually slides towards the (0,0)

point in  $\mathcal{I} - \eta$  space [blue dots in Figs. 4(c)–4(j)] with a concomitant decline in the respective averages in Figs. 5(a)–5(d) as  $g$  is enhanced. For a given finite  $g$ , the crucial difference between the drivers in Eqs. (34) and (35) is captured in the higher variability in the respective means [see Figs. 5(c) and 5(d)] indicating many compatible/equivalent representations chosen by the network for a correlated volume-law entangled state for all sizes. This is true for each individual size of the driver model used, i.e., for each  $N$  and is best illustrated from the plot of the standard deviations (associated with the averages in Fig. 5) displayed in Appendix H, which shows that in c-TFIM at a given  $g$  (especially  $g \rightarrow 0_+$  regime), the trained learner consists of several different  $(k, m)$  pairs with widely varying correlation properties [ $\mathcal{I}(v_k, h_m)$  and  $|\eta(\vec{X})|$  values] arising from compatible chosen for the same learned state. This is much more than the case for TFIM even at a given size  $N$  and given  $g$  value indicating the ability of the learner to distinguish area-law vs volume-law connectivity. However, with increasing  $g$  unanimity sets in as both models displays uncorrelated ground state with unique configurations (see also Appendix H).

Thus in summary the observations indicate three important inferences. Apart from the saturation of LB in  $\mathcal{I}(v_k, h_m)$  vs  $|\eta(\vec{X})|$  space (corroborated for all sizes and many other spin models in Appendix F) as illustrated in Figs. 4(c)–4(j), we also see mirroring of the spin correlation behavior across phase transition between spins of the driver in the correlation introduced between the spins of the visible and latent neurons



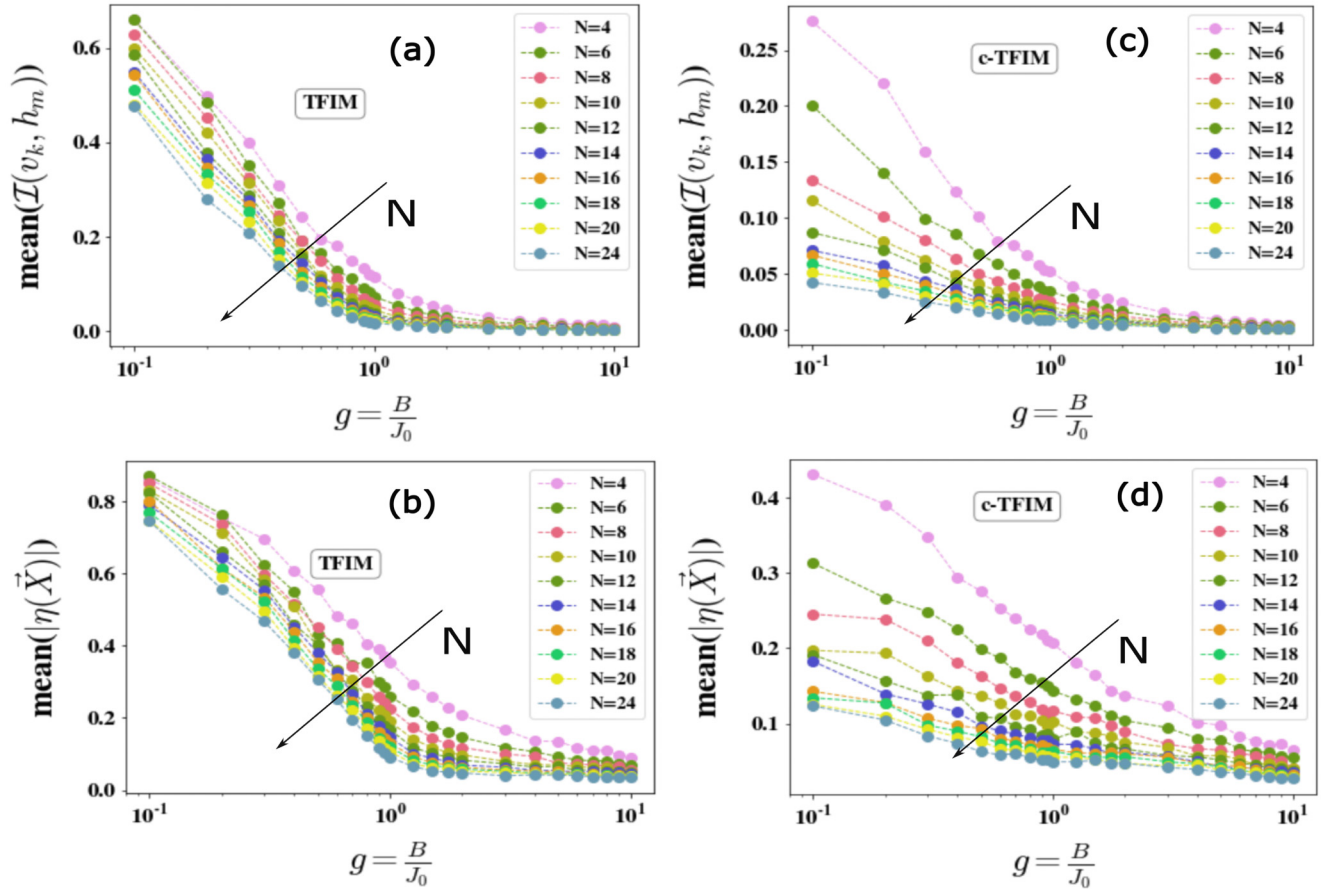


FIG. 5. [(a),(c)] The plot of the mean of  $\mathcal{I}(v_k, h_m)$  vs  $g$  from the likes of Figs. 4(c)–4(j) where the averaging is done over all pairs of  $(k, m)$  and over many initialization at a given  $g$  for various values of driver spins  $N = 4, 6, 8, 10, 12, 14, 16, 18, 20, 24$  for both models defined by Eq. (34) in (a) and Eq. (35) in (c). [(b),(d)] Similar plot as in (a) and (c) but for  $|\eta(\vec{X})|$  as a function of  $g$  for various sizes of driver spins in both the models. In all plots [(a),(d)] transference of spin correlation from both drivers to the visible and latent units of the learner across phase boundaries for drivers of even large sizes is apparent (see also Fig. 11) and so is the higher range of variability of the mean in volume law connected ground states in (c) and (d). It must be emphasized that the mean  $\mathcal{I}(v_k, h_m)$  or mean  $|\eta(\vec{X})|$  [see Eq. (10)] defined is between the visible and the latent/hidden spins. The hidden spins are oblivious to the driver and are only known to the learner network. The configurations of the visible spins of the driver provides a convenient basis to the spins of the driver in both the models. The hidden spins on the other hand are just responsible for supporting the visible spins in adequate feature extraction. Yet, we see in these plots that the spin correlation that exists between the spins/sub-units of the driver across a phase transition is directly transferred or mirrored between the visible and latent spins of the trained learner as desirable features. Such an observation affords a direct numerical quantification to the assistance provided by the hidden spins in training the network  $G = (V, E)$ . This kind of emergent insight into the training of the graph-neural network is central to our paper and as evidenced can easily be procured through the newly constructed  $\mathcal{I}(v_k, h_m)$  vs  $|\eta(\vec{X})|$  space.

of the trained state of the learner  $G$  in Figs. 5(a)–5(d). We also see many equivalent representations of the network for differentiating exotic volume-law connectivity in the driver (see Fig. 5 and Appendix H). It must be emphasized that the last two assertions are true even though the latent neurons are directly oblivious and unrelated to the driver (see the description of the network and how it acts as a variational ansatz for the neural-network encoding of the quantum state as illustrated in Sec. II A). Only the configurations of the visible node register are directly related to the spins of the driver and forms a basis for the eigenspace of the driver. The conventional wisdom is that latent neurons with their respective additional parameters provide support by enhancing the expressibility of the network  $G$ . Our observations thus collectively can serve as a stepping stone towards formalizing and quantifying the important role

performed by the latent spins of the learner  $G$  from a newly obtained perspective of the  $\mathcal{I}(v_k, h_m)$  vs  $|\eta(\vec{X})|$  where physical behavior of such neurons are investigated from the lens of correlation exchange, which happens surreptitiously under the hood during training of  $G$ .

## V. CONCLUSIONS

In this paper, we established a number of key physical insights about the training of the learner network  $G$ . The choice of this specific network is attributed to its astonishing success in simulating a wide variety of quantum systems in condensed matter physics like strongly correlated fermionic assemblies [41,69,70], topologically nontrivial phases [71,72], anyonic symmetries [73], in quantum dynamical evolution [58,74], in

chemistry like 2D-materials [13,44,75] and molecules with multireference correlation [76] under geometric distortion [45], and even in classification tasks [43,77] with quantum or classical data. In fact,  $G$  has been proven to act as a universal approximator for any probability density [35,78] thereby providing a guarantee to the range of its representational capacity. Prior work has also established that the network is capable of mimicking the amplitude of a  $2^n$  dimensional volume-law entangled quantum state even with a sparse representation [79], i.e., using  $O(n)$  parameters as opposed to  $O(np)$  in the usual case. Reference [80] has established that the task of retrieving the full distribution encoded within  $G$  would always entail exponential classical resources unless the polynomial hierarchy collapses. However, quantum circuits to efficiently sample from the same with quadratic qubit and gate costs ( $O(np)$ ) already exist. Another feature enjoyed by the network is its easy extension to  $d$ -dimensional spins [71], which makes encoding higher dimensional quantum states of a driver tractable.

For such a widely recognized network, we have illustrated how changing communication within the sub-units of the learner can be understood by introducing imaginary components of OTOC and have analytically established its relationships with bipartite mutual information. Use of the real part of OTOCs are gaining attention in learning algorithms [81–83] including how quantum learning may be advantageous with such scrambling measures [84], experimental measurement of OTOC on quantum circuits [30,85], bounds on loss function by OTOCs [83], presence of barren plateaus for simulating large scrambling unitaries using parameterized circuits [86]. We see from our analytical deduction that the real part of OTOC between the  $k$ th visible neuron and  $m$ th hidden neuron even though sensitive to  $W_m^k$  through an oscillatory temporal dependence, is completely insensitive to  $(\vec{a}, \vec{b})$  unlike the imaginary part. It must be emphasized that the imaginary-part of OTOC is hitherto unexplored and unreported and is not only new for this network but even for other problems wherein OTOC has been used.

A direct byproduct of our mathematical approach are the discovery of the several conservation theorems/invariants of motion (see Theorem II.1) each of which is equipped with a Lie-algebraic generator that preserves the phase space of OTOC by mapping the the invariants onto itself. We plot both the invariants in Appendix C 1 for a prototypical example of TFIM with  $N = 10$  spins. It is clear from the analysis that only the ones constructed using imaginary components actually change during training. The invariants from real components remain fixed and is insensitive to training epoch. This decisively shows that the importance of imaginary components emphasizing the fact that to procure any information about the learning landscape from these invariants, these components are invincible and the only options available. Such insight into the phase space of OTOC trajectories of network  $G$  and the connection to a hidden Lie-algebraic framework was not procured before and thus provides a richer characterization of the problem, which has escaped prior attention. The mathematical machinery so developed could be used for making similar deductions of OTOC strings in other physical systems and other neural network architectures too.

A direct inter-relationship between the imaginary part of such a four-point correlation function and a two-point correlation function and eventually with mutual information was thereafter deduced. Using the newly framed lens of  $\mathcal{I} - \eta$  space, we have established three different conclusions associated with training the network - saturation of lower bound for a wide variety of spin models, quantifiable transference of spin correlation from the driver to the hidden and visible units of the learner (this is further corroborated using another property computed solely from the learner as demonstrated in Appendix J) and compatible representations chosen by the learner for distinguishing complex connectivity. Such a study begins to probe into the rich underlying universe of the training mechanism and shows that the representations chosen by the learner during the training epoch are quantifiably tuned to readjust the correlation content among the pairwise bi-partitions of visible and hidden neurons commensurate with the changing quantum correlation in the actual driver system. This is despite the fact that the latent spins of the learner are not directly involved in emulating the spins of the driver, they act merely as supportive instruments to accessorize the spins in the visible-node register by enhancing their expressivity.

Applications of such observations may be beneficial to physics-inspired learning [87–89] and also can be leveraged to expedite training through *a priori* informed initialization near lower bound (LB). Extension to explore the ramifications of the findings in classical ML tasks like collaborative filtering and to other important networks like dreaming neural networks [90–92], single-layer autoencoders [93] may be undertaken. Although all results are analytically established, since direct measurement of OTOCs is a possibility even on a quantum circuit [30,31], experimental measurement of the imaginary component of the OTOC string for our learner can be undertaken and its relationship with  $\mathcal{I}$  be exploited. With the present-day promise of machine/deep learning, the authors hope that more such studies will be initiated, which in spirit “humanizes” already-established black-box models by probing into their learning universe, removes the shroud of mystery behind their training, and will hopefully lead to enhanced cross pollination with physical sciences and ameliorated standards in model development. The benefits of such a pipeline can surely be harvested to achieve unprecedented feats in simulating the natural world.

## ACKNOWLEDGMENTS

The authors would like to acknowledge the financial support from the Quantum Science Center, a National Quantum Information Science Research Center of the U.S. Department of Energy (DOE). We also acknowledge funding from National Science Foundation (NSF) under Award No. 1955907.

## APPENDIX A: GENERAL FORMULATION FOR OTOC

As indicated in the text, every OTOC comprises of two unitary operators  $U_1(0), U_2(0)$  chosen usually at two nonlocal sites (here  $\{1, 2\}$  for instance) within a system. The rate at which the information propagates through the

system in real-time is thereafter quantified using the  $C_{U_1, U_2}(t) = \langle U_1^\dagger(0)U_2^\dagger(t)U_1(0)U_2(t) \rangle$ . For measurement purposes, the  $\text{Re}(C_{U_1, U_2}(t))$  is often related in literature to the commutator product given as  $\langle [U_1(0), U_2(t)]^\dagger [U_1(0), U_2(t)] \rangle$  where by virtue of construction of the observable, positive semi-definiteness is ensured. The origin of such a choice is attributed to studying chaotic classical systems wherein an analogous expression for Poisson bracket  $\{x(0), p(t)\}_{PB}$  is used to probe sensitivity to initial conditions. Since in our paper we show that the imaginary part  $\text{Im}(C_{U_1, U_2}(t))$  can also be informative about correlation content within the subunits of the learner  $G$ , we offer herein a general formulation for OTOCs of arbitrary systems using newly constructed positive semidefinite operators involving not only commutators ( $\{\cdot\}$ ) as above but anticommutators ( $\{\cdot\}_+$ ). The advantage of the formulation is the offered generality in the theory of OTOCs in the quantum domain and also the flexibility in projecting both the real and imaginary component of  $C_{U_1, U_2}(t)$  based on user-defined preferences while still maintaining positive semidefiniteness of the observables probed. To this end, let us define a probe as

$$L_{U_1, U_2}(\lambda_1, \lambda_2, t) = \lambda_1 A(U_1, U_2, t) + i\lambda_2 B(U_1, U_2, t), \quad (\text{A1})$$

$$A(U_1, U_2, t) = \{U_1(0), U_2(t)\}_+, \quad (\text{A2})$$

$$B(U_1, U_2, t) = [U_1(0), U_2(t)], \quad (\text{A3})$$

$$\lambda_1, \lambda_2 \in \mathcal{R}. \quad (\text{A4})$$

Using Eq. (A1) and the definition for  $C_{U_1, U_2}(0, 0, \vec{X}, t)$ , it is easy to show

$$\begin{aligned} \text{Re}(C_{U_1, U_2}(0, 0, \vec{X}, t)) &= \text{Re}(\langle U_1^\dagger(0)U_2^\dagger(t)U_1(0)U_2(t) \rangle) \\ &= 1 - \frac{\langle B(U_1, U_2, t)^\dagger B(U_1, U_2, t) \rangle}{2} \quad \because \text{Eq. (A3)} \\ &= 1 - \frac{\langle L_{U_1, U_2}(0, 1, t)^\dagger L_{U_1, U_2}(0, 1, t) \rangle}{2} \quad \because \text{Eq. (A1)} \end{aligned} \quad (\text{A5})$$

and

$$\begin{aligned} \text{Im}(C_{U_1, U_2}(t)) &= \text{Im}(\langle U_1^\dagger(0)U_2^\dagger(t)U_1(0)U_2(t) \rangle) \\ &= -\frac{1}{2} \langle (A(U_1, U_2, t)^\dagger iB(U_1, U_2, t)) \rangle \quad \because \text{Eqs. (A2), (A3)} \\ &= \frac{1}{4} \langle L_{U_1, U_2}(0, 1, t)^\dagger L_{U_1, U_2}(0, 1, t) \rangle \\ &\quad + \frac{1}{4} \langle L_{U_1, U_2}(1, 0, t)^\dagger L_{U_1, U_2}(1, 0, t) \rangle \\ &\quad - \frac{1}{4} \langle L_{U_1, U_2}(1, 1, t)^\dagger L_{U_1, U_2}(1, 1, t) \rangle \quad \because \text{Eq. (A1)}. \end{aligned} \quad (\text{A6})$$

Note that all combinations of  $L_{U_1, U_2}(\lambda_1, \lambda_2, t)^\dagger L_{U_1, U_2}(\lambda_1, \lambda_2, t)$  are positive semidefinite by construction and hence is used in the same vein as the usual commutators for OTOCs are traditionally defined but nonetheless offers a much more general framework for investigating the operator string  $C_{U_1, U_2}(t)$ .

## APPENDIX B: TIME DEPENDENCE OF $\{\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)\}_+$ AND $[\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)]$

To define the time-dependence of  $\{\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)\}_+$  and  $[\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)]$ , it is essential to establish time dependence for  $\sigma^\beta(h_m, t)$  where  $\alpha, \beta \in \{0, x, y, z\}$  with  $\sigma^0 = \mathcal{I}_{2 \times 2}$ . To this end, we prove the following lemma.

*Lemma 1.* For an operator  $\sigma^\beta(h_m, t)$  satisfying the equation  $\dot{\sigma}^\beta(h_m, t) = i[\mathcal{H}, \sigma^\beta(h_m, t)]$ , [generator  $\mathcal{H}$  defined in Eq. (1) in main text], the solution would be  $\sigma^\beta(h_m, t) = e^{2i\mathcal{H}t} \sigma^\beta(h_m, 0) \forall \beta \in \{0, x, y, z\}$  with

$$\mathcal{H}' = \delta_{\beta \in \{x, y\}} (\mathcal{H} - \sum_l a_l \sigma^z(v_l, 0) - \sum_{j \neq m} b_j \sigma^z(h_j, 0) - \sum_{l, j \neq m} W_j^l \sigma^z(v_l, 0) \sigma^z(h_j, 0)).$$

*Proof.*

$$\begin{aligned} \dot{\sigma}^\beta(h_m, t) &= i[\mathcal{H}, \sigma^\beta(h_m, t)] = ie^{i\mathcal{H}t} [\mathcal{H}, \sigma^\beta(h_m, 0)] e^{-i\mathcal{H}t} \\ &= ie^{i\mathcal{H}t} (b_m [\sigma^z(h_m, 0), \sigma^\beta(h_m, 0)] + \sum_l W_m^l [\sigma^z(v_l, 0) \sigma^z(h_m, 0), \sigma^\beta(h_m, 0)]) e^{-i\mathcal{H}t} \\ &= 2i\delta_{\beta \in \{x, y\}} e^{i\mathcal{H}t} (b_m \sigma^z(h_m, 0) \sigma^\beta(h_m, 0) + \sum_l W_m^l \sigma^z(v_l, 0) \sigma^z(h_m, 0) \sigma^\beta(h_m, 0)) e^{-i\mathcal{H}t} \\ &= 2i\delta_{\beta \in \{x, y\}} (b_m \sigma^z(h_m, 0) + \sum_l W_m^l \sigma^z(v_l, 0) \sigma^z(h_m, 0)) \sigma^\beta(h_m, t) \\ &= 2i\delta_{\beta \in \{x, y\}} (\mathcal{H} - \sum_l a_l \sigma^z(v_l, 0) - \sum_{j \neq m} b_j \sigma^z(h_j, 0) - \sum_{l, j \neq m} W_j^l \sigma^z(v_l, 0) \sigma^z(h_j, 0)) \sigma^\beta(h_m, t) = 2i\mathcal{H}' \sigma^\beta(h_m, t), \end{aligned}$$

$$\sigma^\beta(h_m, t) = e^{2i\mathcal{H}'t} \sigma^\beta(h_m, 0). \quad \blacksquare$$

*Lemma 2.* The explicitly time dependant forms of  $[\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)]$  and  $\{\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)\}_+$  are

$$1. \hat{\Theta}_{\alpha, \beta}(t) = [\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)] = -2i \text{Sin}(2W_m^k t) \det \begin{pmatrix} \delta_{\alpha z} & \delta_{\alpha \beta} & \delta_{\alpha \gamma} \\ 1 & \delta_{z\beta} & \delta_{z\gamma} \\ \delta_{\chi z} & \delta_{\chi \beta} & \delta_{\chi \gamma} \end{pmatrix} \hat{\mathcal{F}} \sigma^\chi(v_k, 0) \sigma^\gamma(h_m, 0)$$

$$\begin{aligned}
 2. \hat{\Phi}_{\alpha,\beta}(t) &= \{\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)\}_+ \\
 &= -2i \text{Sin}(2W_m^k t) \det \begin{pmatrix} \delta_{\alpha z} & \delta_{\alpha\beta} & \delta_{\alpha\gamma} \\ 1 & \delta_{z\beta} & \delta_{z\gamma} \\ \delta_{\chi z} & \delta_{\chi\beta} & \delta_{\chi\gamma} \end{pmatrix} \hat{\mathcal{F}} \sigma^\chi(v_k, 0) \sigma^\gamma(h_m, 0) + 2e^{2i\mathcal{H}'t} \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, 0)
 \end{aligned}$$

with  $\hat{\mathcal{F}} = e^{2ib_m\sigma^z(h_m,0)t} \prod_{l \neq k} e^{2iW_m^l \sigma^z(v_l,0)\sigma^z(h_m,0)t}$  being a unitary operator. For definition of  $\mathcal{H}'$  see Lemma 1. Here we restrict  $\alpha, \beta \in \{x, y, z\}$  as the commutation/anticommutation relations with  $\mathcal{I}_{2 \times 2}$  are trivial.

*Proof.*

$$\begin{aligned}
 1. \hat{\Theta}_{\alpha,\beta}(t) &= [\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)] \\
 &= [\sigma^\alpha(v_k, 0), e^{2i\mathcal{H}'t} \sigma^\beta(h_m, 0)] \quad \because \text{Lemma 1} \\
 &= e^{2ib_m\sigma^z(h_m,0)t} \prod_{l \neq k} e^{2iW_m^l \sigma^z(v_l,0)\sigma^z(h_m,0)t} [\sigma^\alpha(v_k, 0), e^{2iW_m^k \sigma^z(v_k,0)\sigma^z(h_m,0)t} \sigma^\beta(h_m, 0)] \\
 &= \hat{\mathcal{F}} [\sigma^\alpha(v_k, 0), i\sigma^z(v_k, 0)\sigma^z(h_m, 0)] \sigma^\beta(h_m, 0) \text{Sin}(2W_m^k t) \\
 &= -2i(\epsilon_{\alpha z \chi})(\epsilon_{z\beta\gamma}) \text{Sin}(2W_m^k t) \hat{\mathcal{F}} \sigma^\chi(v_k, 0) \sigma^\gamma(h_m, 0) \\
 &= -2i \text{Sin}(2W_m^k t) \det \begin{pmatrix} \delta_{\alpha z} & \delta_{\alpha\beta} & \delta_{\alpha\gamma} \\ 1 & \delta_{z\beta} & \delta_{z\gamma} \\ \delta_{\chi z} & \delta_{\chi\beta} & \delta_{\chi\gamma} \end{pmatrix} \hat{\mathcal{F}} \sigma^\chi(v_k, 0) \sigma^\gamma(h_m, 0)
 \end{aligned}$$

$$\begin{aligned}
 2. \hat{\Phi}_{\alpha,\beta}(t) &= \{\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)\}_+ \\
 &= \{\sigma^\alpha(v_k, 0), e^{2i\mathcal{H}'t} \sigma^\beta(h_m, 0)\}_+ \quad \because \text{Lemma 1} \\
 &= [\sigma^\alpha(v_k, 0), e^{2i\mathcal{H}'t}] \sigma^\beta(h_m, 0) + e^{2i\mathcal{H}'t} \{\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, 0)\}_+ \\
 &= [\sigma^\alpha(v_k, 0), e^{2i\mathcal{H}'t}] \sigma^\beta(h_m, 0) + 2e^{2i\mathcal{H}'t} \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, 0) \\
 &= -2i \text{Sin}(2W_m^k t) \det \begin{pmatrix} \delta_{\alpha z} & \delta_{\alpha\beta} & \delta_{\alpha\gamma} \\ 1 & \delta_{z\beta} & \delta_{z\gamma} \\ \delta_{\chi z} & \delta_{\chi\beta} & \delta_{\chi\gamma} \end{pmatrix} \hat{\mathcal{F}} \sigma^\chi(v_k, 0) \sigma^\gamma(h_m, 0) + 2e^{2i\mathcal{H}'t} \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, 0) \\
 &\quad \because \text{Lemma 2(1)}
 \end{aligned}$$

*Lemma 3.* As defined in Lemma 2, if  $\hat{\Theta}_{\alpha,\beta}(t) = [\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)]$  and  $\hat{\Phi}_{\alpha,\beta}(t) = \{\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)\}_+$ , then  $\forall \alpha, \beta \in \{0, x, y, z\}$ ,  $\hat{\Theta}_{\alpha,\beta}(t)$  and  $\hat{\Phi}_{\alpha,\beta}(t)$  satisfies the following operator differential equations: ■

$$\begin{aligned}
 1. \frac{1}{2} \left( \frac{\partial^2 \hat{\Theta}_{\alpha,\beta}(t)^\dagger \hat{\Theta}_{\alpha,\beta}(t)}{\partial t^2} \right) &= \delta_{\alpha \in \{x,y\}} \delta_{\beta \in \{x,y\}} (4W_m^k)^2 \left( \mathcal{I}_{2 \times 2} - \frac{\hat{\Theta}_{\alpha,\beta}(t)^\dagger \hat{\Theta}_{\alpha,\beta}(t)}{2} \right) \\
 2. \frac{\partial^2 \hat{\Phi}_{\alpha,\beta}(t)^\dagger \hat{\Phi}_{\alpha,\beta}(t)}{\partial t^2} &= \delta_{\alpha \in \{x,y\}} \delta_{\beta \in \{x,y\}} (4W_m^k)^2 \hat{\Phi}_{\alpha,\beta}(t)^\dagger \hat{\Theta}_{\alpha,\beta}(t)
 \end{aligned}$$

*Proof.*

(1) Using products like  $\hat{\Theta}_{\alpha,\beta}(t)^\dagger \hat{\Theta}_{\alpha,\beta}(t)$  ensures the unitary operator  $\mathcal{F}$  cancels. The satisfaction of (1) can thereafter be verified explicitly as

$$\begin{aligned}
 \frac{1}{2} \left( \frac{\partial^2 \hat{\Theta}_{\alpha,\beta}(t)^\dagger \hat{\Theta}_{\alpha,\beta}(t)}{\partial t^2} \right) &= 2\delta_{\alpha \in \{x,y\}} \delta_{\beta \in \{x,y\}} (\sigma^\chi(v_k, 0) \sigma^\gamma(h_m, 0))^2 \frac{\partial^2 (\text{Sin}^2(2W_m^k t) \mathcal{F}^\dagger \mathcal{F})}{\partial t^2} \mathcal{I}_{2 \times 2} \quad \because \text{Lemma 2(1)} \\
 &= \delta_{\alpha \in \{x,y\}} \delta_{\beta \in \{x,y\}} (4W_m^k) \frac{\partial \text{Sin}(4W_m^k t)}{\partial t} \mathcal{I}_{2 \times 2} \\
 &= \delta_{\alpha \in \{x,y\}} \delta_{\beta \in \{x,y\}} (4W_m^k)^2 (1 - 2\text{Sin}^2(4W_m^k t)) \mathcal{I}_{2 \times 2} \\
 &= \delta_{\alpha \in \{x,y\}} \delta_{\beta \in \{x,y\}} (4W_m^k)^2 \left( \mathcal{I}_{2 \times 2} - \frac{\hat{\Theta}_{\alpha,\beta}(t)^\dagger \hat{\Theta}_{\alpha,\beta}(t)}{2} \right)
 \end{aligned}$$

(2) Similarly (2) above can be verified explicitly too as follows:

$$\begin{aligned}
 \frac{1}{2} \left( \frac{\partial^2 \hat{\Phi}_{\alpha,\beta}(t)^\dagger \hat{\Theta}_{\alpha,\beta}(t)}{\partial t^2} \right) & \tag{B1} \\
 &= \frac{1}{2} \left( \frac{\partial^2}{\partial t^2} (\hat{\Theta}_{\alpha,\beta}^\dagger(t) \hat{\Theta}_{\alpha,\beta}(t) + 2\sigma^\alpha(v_k, 0) \sigma^\alpha(h_m, 0) e^{-2i\mathcal{H}'t} \hat{\Theta}_{\alpha,\beta}(t)) \right) \quad \because \text{Lemma 2(2)}
 \end{aligned}$$

$$\begin{aligned}
 &= \frac{1}{2} \left( \frac{\partial^2}{\partial t^2} (\hat{\Theta}_{\alpha,\beta}^\dagger(t) \hat{\Theta}_{\alpha,\beta}(t)) \right) + \sigma^\alpha(v_k, 0) \sigma^\alpha(h_m, 0) \left( \frac{\partial^2}{\partial t^2} e^{-2i\mathcal{H}t} \hat{\Theta}_{\alpha,\beta}(t) \right) \\
 &= \frac{1}{2} \left( \frac{\partial^2}{\partial t^2} (\hat{\Theta}_{\alpha,\beta}^\dagger(t) \hat{\Theta}_{\alpha,\beta}(t)) \right) - 2i\epsilon_{\alpha z \chi} \epsilon_{z \beta \gamma} \sigma^\alpha(v_k, 0) \sigma^\alpha(h_m, 0) \\
 &\quad \frac{\partial^2}{\partial t^2} (e^{-2i\mathcal{H}t} \mathcal{F} \sigma^\chi(v_k, 0) \sigma^\gamma(h_m, 0) \sin(2W_m^k t)) \quad \text{see Lemma 2(1) for } \mathcal{F} \\
 &= \frac{1}{2} \left( \frac{\partial^2}{\partial t^2} (\hat{\Theta}_{\alpha,\beta}^\dagger(t) \hat{\Theta}_{\alpha,\beta}(t)) \right) - 2i\epsilon_{\alpha z \chi} \epsilon_{z \beta \gamma} \sigma^\alpha(v_k, 0) \sigma^\alpha(h_m, 0) \\
 &\quad \frac{\partial^2}{\partial t^2} (\sigma^\chi(v_k, 0) \sigma^\gamma(h_m, 0) \cos(W_m^k t) \sin(2W_m^k t) + \epsilon_{z \chi \omega} \epsilon_{z \gamma \eta} \sigma^\omega(v_k, 0) \sigma^\eta(h_m, 0) \sin^2(2W_m^k t)) \\
 &= \frac{1}{2} \left( \frac{\partial^2}{\partial t^2} (\hat{\Theta}_{\alpha,\beta}^\dagger(t) \hat{\Theta}_{\alpha,\beta}(t)) \right) - 2i\epsilon_{\alpha z \chi} \epsilon_{z \beta \gamma} \sigma^\alpha(v_k, 0) \sigma^\alpha(h_m, 0) \\
 &\quad \frac{\partial^2}{\partial t^2} (\sigma^\chi(v_k, 0) \sigma^\gamma(h_m, 0) \cos(W_m^k t) \sin(2W_m^k t)) - \frac{1}{2} \left( \frac{\partial^2}{\partial t^2} (\hat{\Theta}_{\alpha,\beta}^\dagger(t) \hat{\Theta}_{\alpha,\beta}(t)) \right) \\
 &= -2i\epsilon_{\alpha z \chi} \epsilon_{z \beta \gamma} \sigma^\alpha(v_k, 0) \sigma^\alpha(h_m, 0) \frac{\partial^2}{\partial t^2} (\sigma^\chi(v_k, 0) \sigma^\gamma(h_m, 0) \cos(W_m^k t) \sin(2W_m^k t)) \\
 &= 2i\epsilon_{\alpha z \chi} \epsilon_{z \beta \gamma} \epsilon_{\alpha \chi \kappa} \epsilon_{\beta \gamma \omega} \frac{\partial^2}{\partial t^2} (\sigma^\kappa(v_k, 0) \sigma^\omega(h_m, 0) \cos(W_m^k t) \sin(2W_m^k t)) \\
 &= -2i \frac{\partial^2}{\partial t^2} (\sigma^z(v_k, 0) \sigma^z(h_m, 0) \cos(W_m^k t) \sin(2W_m^k t)) \\
 &= \frac{1}{2} \frac{\partial^2}{\partial t^2} (-2i \sigma^z(v_k, 0) \sigma^z(h_m, 0) \sin(4W_m^k t)) \\
 &= -(4W_m^k)^2 \hat{\Phi}_{\alpha,\beta}(t)^\dagger \hat{\Theta}_{\alpha,\beta}(t), \tag{B2}
 \end{aligned}$$

where the last equality follows from Eqs. (B1) and (B2), which shows  $\frac{1}{2} \left( \frac{\partial^2 \hat{\Phi}_{\alpha,\beta}(t)^\dagger \hat{\Theta}_{\alpha,\beta}(t)}{\partial t^2} \right) = \frac{1}{2} \frac{\partial^2}{\partial t^2} (-2i \sigma^z(v_k, 0) \sigma^z(h_m, 0) \sin(4W_m^k t))$  thereby implying  $\hat{\Phi}_{\alpha,\beta}(t)^\dagger \hat{\Theta}_{\alpha,\beta}(t) = -2i \sigma^z(v_k, 0) \sigma^z(h_m, 0) \sin(4W_m^k t)$ . ■

**APPENDIX C: INVARIANTS OF MOTION-PROOF OF THEOREM 1 IN MAIN TEXT**

We are now in a position to prove the assertions of Theorem 1 in the main text  $\forall \alpha, \beta \in \{x, y\}$ .

(1) To establish the assertions of Theorem 1 (1) we need to prove the following primitive lemmas first for completeness.

*Lemma 4.*  $\sigma^\alpha f(\sigma^z) (\sigma^\alpha)^\dagger = f(-\sigma^z) \quad \forall \alpha \in \{x, y\}$

*Proof.*

$$\begin{aligned}
 \sigma^\alpha \sigma^z (\sigma^\alpha)^\dagger &= i \sigma^\alpha \epsilon_{z \alpha k} \sigma^k = i^2 \epsilon_{z \alpha k} \epsilon_{\alpha k \omega} \sigma^\omega = i^2 \delta_{\omega z} \sigma^\omega = -\sigma^z \quad \because (\alpha \in \{x, y\}) \\
 \sigma^\alpha f(\sigma^z) (\sigma^\alpha)^\dagger &= \sigma^\alpha [\beta_0 + \beta_1 \sigma^z + \beta_2 (\sigma^z)^2 + \dots] (\sigma^\alpha)^\dagger \\
 &= \beta_0 \sigma^\alpha (\sigma^\alpha)^\dagger + \beta_1 \sigma^\alpha \sigma^z (\sigma^\alpha)^\dagger + \beta_2 [\sigma^\alpha \sigma^z (\sigma^\alpha)^\dagger]^2 + \dots \\
 &= \beta_0 - \beta_1 \sigma^z + \beta_2 (\sigma^z)^2 - \beta_3 (\sigma^z)^3 + \dots \\
 &= f(-\sigma^z)
 \end{aligned}$$

*Lemma 5.* For a diagonal matrix A and an off-diagonal matrix B,  $\text{Tr}(AB) = 0$ .

Given:  $B_i^k = 0 \quad \forall i = k$  and  $A_i^k = a_i \delta_i^k$

*Proof.*

$$\text{Tr}(AB) = \sum_i (AB)_i^i = \sum_i A_i^i B_i^i = \sum_i a_i \delta_i^i B_i^i = \sum_i a_i B_i^i = 0.$$

*Theorem C.1.* [Theorem II.1(1) in main text] For a given parameter vector  $\vec{X}$ , one can define  $\mathcal{H}(\vec{X}, \vec{v}, \vec{h})$  [see Eq. (2)] and a thermal state  $\rho_{th}(\vec{X}, \vec{v}, \vec{h})$ . Let us thereafter define the following OTOC with  $U_1(0) = \tilde{\sigma}_\alpha = \sigma^\alpha(v_k, 0) - \kappa_1 \mathcal{I}$ , and operator  $U_2(0) = \tilde{\sigma}_\beta = \sigma^\beta(h_m, 0) - \kappa_2 \mathcal{I}$  and the generator  $H_{otoc} = \mathcal{H}(\vec{X}, \vec{v}, \vec{h})$  [in Eq. (5) in main text]  $\forall \{\alpha, \beta\} \in \{x, y\}$ .

$$C_{\sigma^\alpha, \tilde{\sigma}^\beta}(\kappa_1, \kappa_2, \vec{X}, t) = \langle \tilde{\sigma}^\alpha(v_k, 0) \tilde{\sigma}^\beta(h_m, t) \tilde{\sigma}^\alpha(v_k, 0) \tilde{\sigma}^\beta(h_m, t) \rangle. \tag{C1}$$

Note that  $\{\kappa_1, \kappa_2\} \in \mathcal{C}^2$  are arbitrary user-defined mean translations. Also  $\langle \cdot \rangle$  indicates averaging over the thermal state  $\rho_{th}(\vec{X}, \vec{v}, \vec{h})$ , which activates the  $\vec{X}$  dependence. Using 6, one can then make the following statements: For  $(\kappa_1, \kappa_2) \in \mathcal{C}^2$ ,

$$C_{\sigma^\alpha, \sigma^\beta}(\kappa_1, \kappa_2, \vec{X}, t) = \langle \tilde{\sigma}^\alpha(v_k, 0) \tilde{\sigma}^\beta(h_m, t) \tilde{\sigma}^\alpha(v_k, 0) \tilde{\sigma}^\beta(h_m, t) \rangle = C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, t) + |\kappa_1|^2 |\kappa_2|^2 + |\kappa_2|^2 + |\kappa_1|^2$$

where  $\alpha, \beta \in \{x, y\}$ .

*Proof.* Let us define  $\tilde{U}_1(0) = U_1(0) - \kappa_1 \mathcal{I}$  and  $\tilde{U}_2(0) = U_2(0) - \kappa_2 \mathcal{I}$  where  $U_1(0)$  and  $U_2(0)$  are two unitary operators as defined in the main text

$$\begin{aligned} \langle \tilde{U}_1(0)^\dagger \tilde{U}_2(t)^\dagger \tilde{U}_1(0) \tilde{U}_2(t) \rangle &= \langle U_1(0)^\dagger U_2(t)^\dagger U_1(0) U_2(t) \rangle + |\kappa_1|^2 |\kappa_2|^2 - \kappa_2 \langle U_1(0)^\dagger U_2(t)^\dagger U_1(0) \rangle - \kappa_1 \langle U_1(0)^\dagger U_2(t)^\dagger U_2(t) \rangle \\ &\quad - \kappa_2^* \langle U_1(0)^\dagger U_1(0) U_2(t) \rangle - \kappa_1^* \langle U_2(t)^\dagger U_1(0) U_2(t) \rangle + |\kappa_2|^2 \langle U_1^\dagger(0) U_1(0) \rangle \\ &\quad + |\kappa_1|^2 \langle U_2^\dagger(t) U_2(t) \rangle + \kappa_1^* \kappa_2 \langle U_2^\dagger(t) U_1(0) \rangle + \kappa_1 \kappa_2^* \langle U_1^\dagger(0) U_2(t) \rangle \\ &\quad + \kappa_1^* \kappa_2^* \langle U_1(0) U_2(t) \rangle + \kappa_1 \kappa_2 \langle U_1^\dagger(0) U_2^\dagger(t) \rangle - \kappa_1 |\kappa_2|^2 \langle U_1(0)^\dagger \rangle \\ &\quad - |\kappa_1|^2 \kappa_2 \langle U_2(t)^\dagger \rangle - \kappa_1^* |\kappa_2|^2 \langle U_1(0) \rangle - |\kappa_1|^2 \kappa_2^\dagger \langle U_2(t) \rangle. \end{aligned} \quad (C2)$$

Let us now substitute  $U_1(0) = \sigma^\alpha(v_k, 0)$  and  $U_2(t) = \sigma^\beta(h_m, t)$  in Eq. (C2).

By definition

$$\langle \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, t) \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, t) \rangle = C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, t). \quad (C3)$$

Also one can show the following:

(a)

$$\begin{aligned} \langle \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, t) \sigma^\alpha(v_k, 0) \rangle &= \text{Tr}(\rho_{th}(\vec{X}, v_k, h_m) \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, t) \sigma^\alpha(v_k, 0)) \\ &= \text{Tr}(\sigma^\alpha(v_k, 0) \rho_{th}(\vec{X}, v_k, h_m) \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, t)) \\ &= \text{Tr}(\rho_{th}(\vec{X}, -v_k, h_m) e^{i\mathcal{H}t} \sigma^\beta(h_m, 0) e^{-i\mathcal{H}t}) \quad \text{see Lemma 4 and Lemma 1} \\ &= \text{Tr}(e^{-i\mathcal{H}t} \rho_{th}(\vec{X}, -v_k, h_m) e^{i\mathcal{H}t} \sigma^\beta(h_m, 0)) \\ &= \text{Tr}(\rho_{th}(\vec{X}, -v_k, h_m) \sigma^\beta(h_m, 0)) \quad \because e^{i\mathcal{H}t} \& \rho \text{ are diagonal} \\ &= 0 \quad \text{see Lemma 5} \end{aligned} \quad (C4)$$

(b)

$$\begin{aligned} \langle \sigma^\beta(h_m, t) \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, t) \rangle &= \text{Tr}(\rho_{th}(\vec{X}, v_k, h_m) \sigma^\beta(h_m, t) \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, t)) \\ &= \text{Tr}(\sigma^\beta(h_m, t) \rho_{th}(\vec{X}, v_k, h_m) \sigma^\beta(h_m, t) \sigma^\alpha(v_k, 0)) \\ &= \text{Tr}(e^{i\mathcal{H}t} \sigma^\beta(h_m, 0) e^{-i\mathcal{H}t} \rho_{th}(\vec{X}, v_k, h_m) e^{i\mathcal{H}t} \sigma^\beta(h_m, 0) e^{-i\mathcal{H}t} \sigma^\alpha(v_k, 0)) \\ &= \text{Tr}(e^{i\mathcal{H}t} \sigma^\beta(h_m, 0) \rho_{th}(\vec{X}, v_k, h_m) \sigma^\beta(h_m, 0) e^{-i\mathcal{H}t} \sigma^\alpha(v_k, 0)) \quad \because e^{i\mathcal{H}t} \& \rho \text{ are diagonal} \\ &= \text{Tr}(e^{i\mathcal{H}t} \rho_{th}(\vec{X}, v_k, -h_m) e^{-i\mathcal{H}t} \sigma^\alpha(v_k, 0)) \quad \text{see Lemma 4} \\ &= \text{Tr}(\rho_{th}(\vec{X}, v_k, -h_m) \sigma^\alpha(v_k, 0)) \quad \because e^{i\mathcal{H}t} \& \rho \text{ are diagonal} \\ &= 0 \quad \text{see Lemma 5} \end{aligned} \quad (C5)$$

(c)

$$\langle \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, t) \sigma^\beta(h_m, t) \rangle = \langle \sigma^\alpha(v_k, 0) \rangle = \text{Tr}(\rho_{th}(\vec{X}, v_k, h_m) \sigma^\alpha(v_k, 0)) \quad \text{see Lemma 5} \quad (C6)$$

(d)

$$\begin{aligned} \langle \sigma^\alpha(v_k, 0) \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, t) \rangle &= \langle \sigma^\beta(h_m, t) \rangle = \text{Tr}(\rho_{th}(\vec{X}, v_k, h_m) \sigma^\beta(h_m, t)) \\ &= 0 \quad \text{see Lemma 5} \end{aligned} \quad (C7)$$

(e)

$$\begin{aligned} \langle \sigma^\beta(h_m, t) \sigma^\alpha(v_k, 0) \rangle &= \text{Tr}(\rho_{th}(\vec{X}, v_k, h_m) \sigma^\beta(h_m, t) \sigma^\alpha(v_k, 0)) = \text{Tr}(\rho_{th}(\vec{X}, v_k, h_m) e^{2i\mathcal{H}t} \sigma^\beta(h_m, 0) \sigma^\alpha(v_k, 0)) \quad \text{see Lemma 1} \\ &= 0 \quad \text{see Lemma 5} \end{aligned} \quad (C8)$$

Here  $\rho e^{2i\mathcal{H}t}$  is diagonal,  $\sigma^\beta(h_m, 0) \sigma^\alpha(v_k, 0)$  is off-diagonal.

Similarly we can show  $\langle \sigma^\alpha(v_k, 0) \sigma^\beta(h_m, t) \rangle = 0$ ,

$\langle \sigma^\alpha(v_k, 0) \rangle = \langle \sigma^\beta(h_m, t) \rangle = 0$  see Lemma 5

Substituting the results of Eqs. (C3)–(C8) in Eq. (C2) establishes the claim of the theorem. ■

(2) Now we establish the assertions of Theorem II.1(2) in the main text.

*Theorem C.2.* [Theorem II.1(2) in main text] For a given parameter vector  $\vec{X}$ , one can define  $\mathcal{H}(\vec{X}, \vec{v}, \vec{h})$  [see Eq. (2)] and a thermal state  $\rho_{th}(\vec{X}, \vec{v}, \vec{h})$ . Let us thereafter define the following OTOC with  $U_1(0) = \tilde{\sigma}_\alpha = \sigma^\alpha(v_k, 0) - \kappa_1 \mathcal{I}$ , and operator  $U_2(0) = \tilde{\sigma}_\beta = \sigma^\beta(h_m, 0) - \kappa_2 \mathcal{I}$  and the generator  $H_{otoc} = \mathcal{H}(\vec{X}, \vec{v}, \vec{h})$  [in Eq. (5) in main text]  $\forall \{\alpha, \beta\} \in \{x, y\}$ .

$$C_{\sigma^\alpha, \sigma^\beta}(\kappa_1, \kappa_2, \vec{X}, t) = \langle \tilde{\sigma}^\alpha(v_k, 0) \tilde{\sigma}^\beta(h_m, t) \tilde{\sigma}^\alpha(v_k, 0) \tilde{\sigma}^\beta(h_m, t) \rangle. \tag{C9}$$

Note that  $\{\kappa_1, \kappa_2\} \in \mathcal{C}^2$  are arbitrary user-defined mean translations. Also  $\langle \cdot \rangle$  indicates averaging over the thermal state  $\rho_{th}(\vec{X}, \vec{v}, \vec{h})$ , which activates the  $\vec{X}$  dependence. Using 6, one can then show the following invariants of motion exists for  $C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, t)$ :

$$\begin{aligned} (1) \quad I_1 &= -2\dot{\xi}_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) \text{Cos}(\tau) - 2\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) \text{Sin}(\tau) \\ (2) \quad I_2 &= -2\dot{\xi}_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) \text{Sin}(\tau) + 2\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) \text{Cos}(\tau) \end{aligned}$$

where  $\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau)$  can either be the real or the imaginary part of  $(C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau))$  and  $\dot{\xi}$  is  $\frac{d\xi}{d\tau}$  with  $\tau = 4W_m^k t$

**Real part**

For the real part of  $C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)$  we substitute in Theorem C.2 (1)

$$\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) = \text{Re}(C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)) = C_{r, \sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau). \tag{C10}$$

Thus we have

(a)

$$I_{r1} = -2\dot{C}_{r, \sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) \text{cos}(\tau) - 2C_{r, \sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) \text{sin}(\tau). \tag{C11}$$

*Proof.* To prove  $I_{r1}$  as an invariant, we have to show that  $\dot{I}_{r1} = 0$ . Using the definition of  $C_{r, \sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)$  as

$$C_{r, \sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) = \langle \mathcal{I}_{2 \times 2} - \frac{1}{2} \hat{\Theta}_{\alpha, \beta}^\dagger(\tau) \hat{\Theta}_{\alpha, \beta}(\tau) \rangle_{\rho_{th}}, \tag{C12}$$

where  $\hat{\Theta}_{\alpha, \beta}(t) = [\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)]$  (see Lemma 2) one can show the following:

$$\begin{aligned} \dot{I}_{r1} &= \left\langle \frac{\partial^2 \hat{\Theta}_{\alpha, \beta}^\dagger(\tau) \hat{\Theta}_{\alpha, \beta}(\tau)}{\partial \tau^2} \text{Cos}(\tau) + \frac{\partial \hat{\Theta}_{\alpha, \beta}^\dagger(\tau) \hat{\Theta}_{\alpha, \beta}(\tau)}{\partial \tau} \text{Sin}(\tau) - \frac{\partial \hat{\Theta}_{\alpha, \beta}^\dagger(\tau) \hat{\Theta}_{\alpha, \beta}(\tau)}{\partial t} \text{Sin}(\tau) \right. \\ &\quad \left. - 2(\mathcal{I}_{2 \times 2} - \frac{1}{2} \hat{\Theta}_{\alpha, \beta}^\dagger(\tau) \hat{\Theta}_{\alpha, \beta}(\tau)) \text{Cos}(\tau) \right\rangle_{\rho_{th}} \\ &= \left\langle \frac{\partial^2 \hat{\Theta}_{\alpha, \beta}^\dagger(\tau) \hat{\Theta}_{\alpha, \beta}(\tau)}{\partial \tau^2} \text{Cos}(\tau) - 2 \left( \mathcal{I}_{2 \times 2} - \frac{1}{2} \hat{\Theta}_{\alpha, \beta}^\dagger(t) \hat{\Theta}_{\alpha, \beta}(\tau) \right) \text{Cos}(\tau) \right\rangle_{\rho_{th}} \\ &= 2 \left\langle \frac{1}{2} \frac{\partial^2 \hat{\Theta}_{\alpha, \beta}^\dagger(\tau) \hat{\Theta}_{\alpha, \beta}(\tau)}{\partial \tau^2} - \left( \mathcal{I}_{2 \times 2} - \frac{1}{2} \hat{\Theta}_{\alpha, \beta}^\dagger(t) \hat{\Theta}_{\alpha, \beta}(\tau) \right) \right\rangle_{\rho_{th}} \text{Cos}(\tau) \\ &= \frac{2}{(4W_m^k)^2} \left\langle \frac{1}{2} \frac{\partial^2 \hat{\Theta}_{\alpha, \beta}^\dagger(t) \hat{\Theta}_{\alpha, \beta}(t)}{\partial t^2} - (4W_m^k)^2 \left( \mathcal{I}_{2 \times 2} - \frac{1}{2} \hat{\Theta}_{\alpha, \beta}^\dagger(t) \hat{\Theta}_{\alpha, \beta}(t) \right) \right\rangle_{\rho_{th}} \text{Cos}(4W_m^k t) \quad \because \tau = 4W_m^k t \\ &= 0 \quad \because \text{Lemma 3(1)} \quad \forall \alpha, \beta \in \{x, y\} \end{aligned}$$

(b) For this one we have to substitute in in Theorem C.2 (2) ■

$$\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) = \text{Re}(C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)) = C_{r, \sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) \tag{C13}$$

After substitution, we have

$$I_{r2} = -2\dot{C}_{r, \sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) \text{sin}(\tau) + 2C_{r, \sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) \text{cos}(\tau) \tag{C14}$$

*Proof.* As before to prove invariance one has to show  $\dot{I}_{r2} = 0$ . Similar to Eq. (C11), Eq. (C14) can either be verified explicitly or by noting  $I_{r2} = I_{r1}(\tau \rightarrow \tau - \frac{\pi}{2})$ . The invariance of  $I_{r2}$  thereafter follows from the invariance of  $I_{r1}$  proven above. ■

**Imag part**

For the imaginary part of  $C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)$  we substitute in Theorem C.2 (1)

$$\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) = \text{Im}(C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)) = C_{i, \sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) \tag{C15}$$

Thus we have now

(a)

$$I_{i1} = -2\dot{C}_{i,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)\cos(\tau) - 2C_{i,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)\sin(\tau) \quad (C16)$$

*Proof.* To prove  $I_{i1}$  as an invariant, we have to show that  $\dot{I}_{i1} = 0$ . Using the definition of  $C_{i,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)$  as

$$C_{i,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau) = -\frac{i}{2}\langle \hat{\Phi}_{\alpha,\beta}^\dagger(t)\hat{\Theta}_{\alpha,\beta}(\tau) \rangle_{\rho_{th}} \quad (C17)$$

where  $\hat{\Phi}_{\alpha,\beta}(t) = \{\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)\}_+$  (see Lemma 2), one can show the following:

$$\begin{aligned} \dot{I}_{i1} &= i \left\langle \frac{\partial^2 \hat{\Phi}_{\alpha,\beta}^\dagger(\tau)\hat{\Theta}_{\alpha,\beta}(\tau)}{\partial \tau^2} \cos(\tau) - \frac{\partial \hat{\Phi}_{\alpha,\beta}^\dagger(\tau)\hat{\Theta}_{\alpha,\beta}(\tau)}{\partial \tau} \sin(\tau) + \frac{\partial \hat{\Phi}_{\alpha,\beta}^\dagger(\tau)\hat{\Theta}_{\alpha,\beta}(\tau)}{\partial \tau} \sin(\tau) + \hat{\Phi}_{\alpha,\beta}^\dagger(\tau)\hat{\Theta}_{\alpha,\beta}(\tau)\cos(\tau) \right\rangle_{\rho_{th}} \\ &= i \left\langle \frac{\partial^2 \hat{\Phi}_{\alpha,\beta}^\dagger(\tau)\hat{\Theta}_{\alpha,\beta}(\tau)}{\partial \tau^2} \cos(\tau) + \hat{\Phi}_{\alpha,\beta}^\dagger(\tau)\hat{\Theta}_{\alpha,\beta}(\tau)\cos(\tau) \right\rangle_{\rho_{th}} \\ &= \frac{i}{(4W_m^k)^2} \left\langle \frac{\partial^2 \hat{\Phi}_{\alpha,\beta}^\dagger(t)\hat{\Theta}_{\alpha,\beta}(t)}{\partial t^2} \cos(4W_m^k t) + (4W_m^k)^2 \hat{\Phi}_{\alpha,\beta}^\dagger(t)\hat{\Theta}_{\alpha,\beta}(t)\cos(4W_m^k t) \right\rangle_{\rho_{th}} \quad \because \tau = 4W_m^k t \\ &= 0 \quad \because \text{Lemma 3(2)} \quad \forall \alpha, \beta \in \{x, y\} \end{aligned}$$

■

(b) For this one we substitute in Theorem C.2 (2)

$$\xi_{\sigma^\alpha,\sigma^\beta}(\vec{X}, \tau) = \text{Im}(C_{\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)) = C_{i,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau). \quad (C18)$$

Thus after substitution we have

$$I_{i2} = -2\dot{C}_{i,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)\sin(\tau) + 2C_{i,\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)\cos(\tau) \quad (C19)$$

*Proof.* Similar to Eq. (C16), (C19) can either be verified explicitly or by noting  $I_{i2} = I_{i1}(\tau \rightarrow \tau - \frac{\pi}{2})$ . The invariance of  $I_{i2}$  thereafter follows from the invariance of  $I_{i1}$  proven above. ■

Using the definitions for  $\Theta_{\alpha,\beta,\tau} = [\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)]$  and  $\Phi_{\alpha,\beta,\tau} = \{\sigma^\alpha(v_k, 0), \sigma^\beta(h_m, t)\}_+$  as given in Lemma 2, one can enlist the initial conditions satisfied by  $C_{r\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)$  and  $C_{r\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau)$  as follows:

$$C_{r\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, 0) = \left\langle \mathcal{I}_{2 \times 2} - \frac{1}{2}\hat{\Theta}_{\alpha,\beta}^\dagger(0)\hat{\Theta}_{\alpha,\beta}(0) \right\rangle_{\rho_{th}} = 1 \quad (\because \hat{\Theta}_{\alpha,\beta}(0) = 0), \quad (C20)$$

$$\dot{C}_{r\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, 0) = -\frac{1}{2} \left\langle \frac{\partial}{\partial \tau} \hat{\Theta}_{\alpha,\beta}^\dagger(\tau)\hat{\Theta}_{\alpha,\beta}(\tau) \right\rangle_{\rho_{th}} \Big|_0 = 0 \quad (\because \hat{\Theta}_{\alpha,\beta}(0) = \hat{\Theta}_{\alpha,\beta}^\dagger(0) = 0), \quad (C21)$$

$$C_{i\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, 0) = -\frac{i}{2} \langle \hat{\Phi}_{\alpha,\beta}^\dagger(0)\hat{\Theta}_{\alpha,\beta}(0) \rangle_{\rho_{th}} = 0 \quad (\because \hat{\Theta}_{\alpha,\beta}(0) = 0), \quad (C22)$$

$$\begin{aligned} \dot{C}_{i\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, 0) &= -\frac{i}{2} \left\langle \frac{\partial}{\partial \tau} \hat{\Phi}_{\alpha,\beta}^\dagger(\tau)\hat{\Theta}_{\alpha,\beta}(\tau) \right\rangle_{\rho_{th}} \Big|_0 = \left\langle \frac{\partial}{\partial \tau} \sigma^z(v_k, 0)\sigma^z(h_m, 0)\sin(4W_m^k t) \right\rangle_{\rho_{th}} \Big|_0 \\ &= \langle \sigma^z(v_k, 0)\sigma^z(h_m, 0) \rangle_{\rho_{th}} \quad (\text{see Lemma 2(2)}). \end{aligned} \quad (C23)$$

Using Eqs. (C20)–(C23) one can establish the following corollary to Theorem 1 in the main text.

*Corollary 5.1.* The following statements are true:

(i)  $C_{r\sigma^\alpha,\sigma^\beta}(0, 0, \vec{X}, \tau) = \text{Cos}(4W_m^k t)$



*Proof.*

$$C_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) = \frac{I_{r2}}{2}\text{Cos}(\tau) + \frac{I_{r1}}{2}\text{Sin}(\tau) \quad (\because \text{Eq. (C11)} \times \text{Sin}(\tau) - \text{Eq. (C14)} \times \text{Cos}(\tau))$$

$$= \text{Cos}(\tau) \quad (\text{see Eq. C20, C21}) = \text{Cos}(4W_m^k t) \quad \because \tau = 4W_m^k t$$

$$C_{i\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) = \langle \sigma^z(v_k, 0) \sigma^z(h_m, 0) \rangle \text{Sin}(4W_m^k t)$$

*Proof.*

$$(ii) C_{i\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) = \frac{I_{i2}}{2}\text{Cos}(\tau) + \frac{I_{i1}}{2}\text{Sin}(\tau) \quad (\because \text{Eq. (C16)} \times \text{Sin}(\tau) - \text{Eq. (C19)} \times \text{Cos}(\tau))$$

$$= \langle \sigma^z(v_k, 0) \sigma^z(h_m, 0) \rangle \text{Sin}(\tau) \quad (\text{see Eq. C22, C23}) = \langle \sigma^z(v_k, 0) \sigma^z(h_m, 0) \rangle \text{Sin}(4W_m^k t) \quad (\because \tau = 4W_m^k t)$$

*Corollary 5.2.* In addition to the above invariants, following are also the invariants of motion for  $C_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)$  and  $C_{i\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)$

$$(i) (\dot{C}_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)^2 - C_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)^2)\text{Sin}(2\tau) - 2\dot{C}_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)C_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)\text{Cos}(2\tau)$$

*Proof.* Can be verified through explicit evaluation using the solution  $C_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)$  in Corollary 5.1, or by combining the invariants  $I_{r1}, I_{r2}$  in Eqs. (C11) and (C14) as  $\frac{1}{2}I_{r1}I_{r2}$ . This invariant is plotted in Fig. 6 for the real part. Exactly similar profile for the invariant exists for the imaginary part too. ■

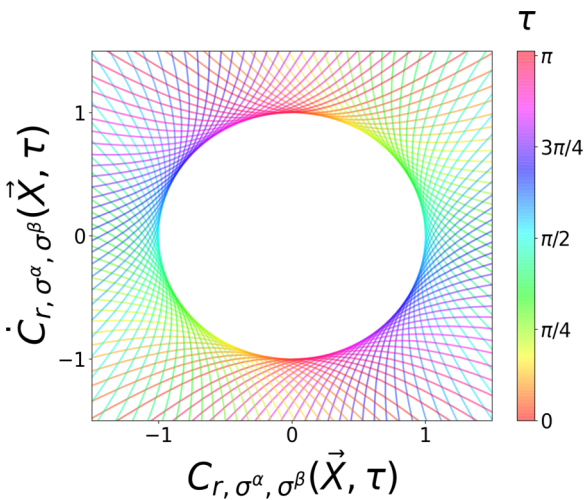


FIG. 6. The profile for the real part of the invariant  $\frac{I_{r1}I_{r2}}{2}$  (see Appendix C, Corollary 5.2), where  $C_{r, \sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) = \text{Re}(C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)) = \xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau)$  and  $\alpha = \beta = x$  is substituted in Theorem 1(2) in main text for a specific vis-hid neuron pair  $(k, m)$  [see Eq. (6) in main text]. Subscript “r” denotes the real part in the plot. The quantity has been evaluated at  $(\kappa_1 = 0, \kappa_2 = 0)$ , hence the explicit dependence on  $(\kappa_1, \kappa_2)$  has been dropped for notational brevity. Each hyperbolic curve is the loci of points with the same fixed value of the invariant and a particular fixed value of  $\tau = 4W_m^k t$  as the dependant variable (see colorbar). The loci of all points touching several hyperbolic curves with a fixed value of the invariant alone but wherein  $\tau$  is continuously changed  $\in [0, \pi]$  forms a circle at the center. Since  $\tau = 4W_m^k t$ , this change can be administered by changing real-time ( $t$ ) or the parameter vector  $\vec{X}$  during training which affects  $W_m^k$ . Exactly similar profile exists for the imaginary part too where  $C_{i, \sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) = \text{Im}(C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)) = \xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau)$  is substituted.

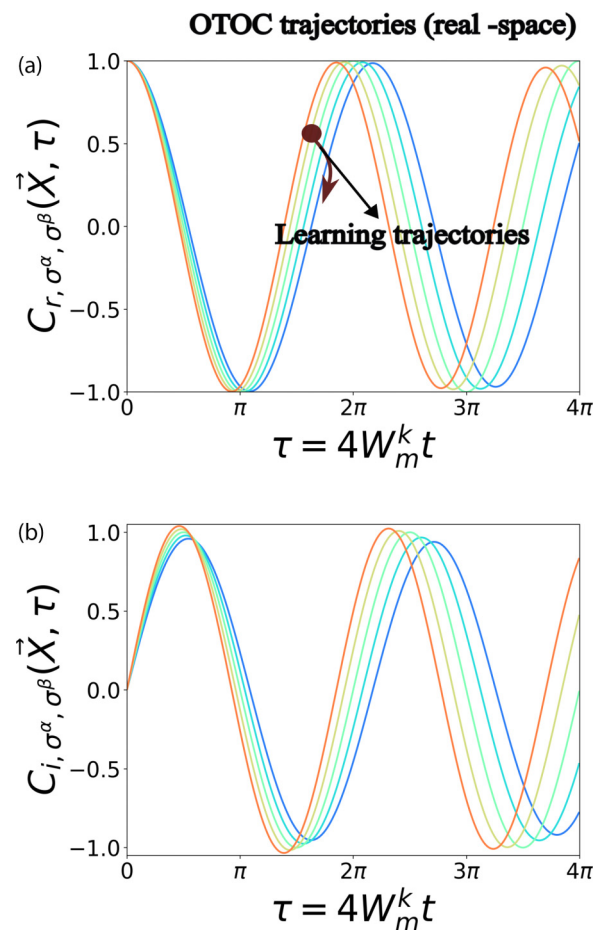


FIG. 7. (a) The harmonic behavior of  $C_{r, \sigma^\alpha, \sigma^\beta}(\vec{X}, \tau)$  in real-time with the frequency equal to  $4W_m^k$ . Note while training  $G$ , a typical learning trajectory in parameter space  $\vec{X}$  amounts to hopping from one such curve to another as indicated. (b) The harmonic behavior of  $C_{i, \sigma^\alpha, \sigma^\beta}(\vec{X}, \tau)$  in real-time with the same frequency as in (a), i.e.,  $4W_m^k$  but phase-shifted from (a) by  $\frac{\pi}{2}$  (refer to Appendix C, Corollary 5.1).

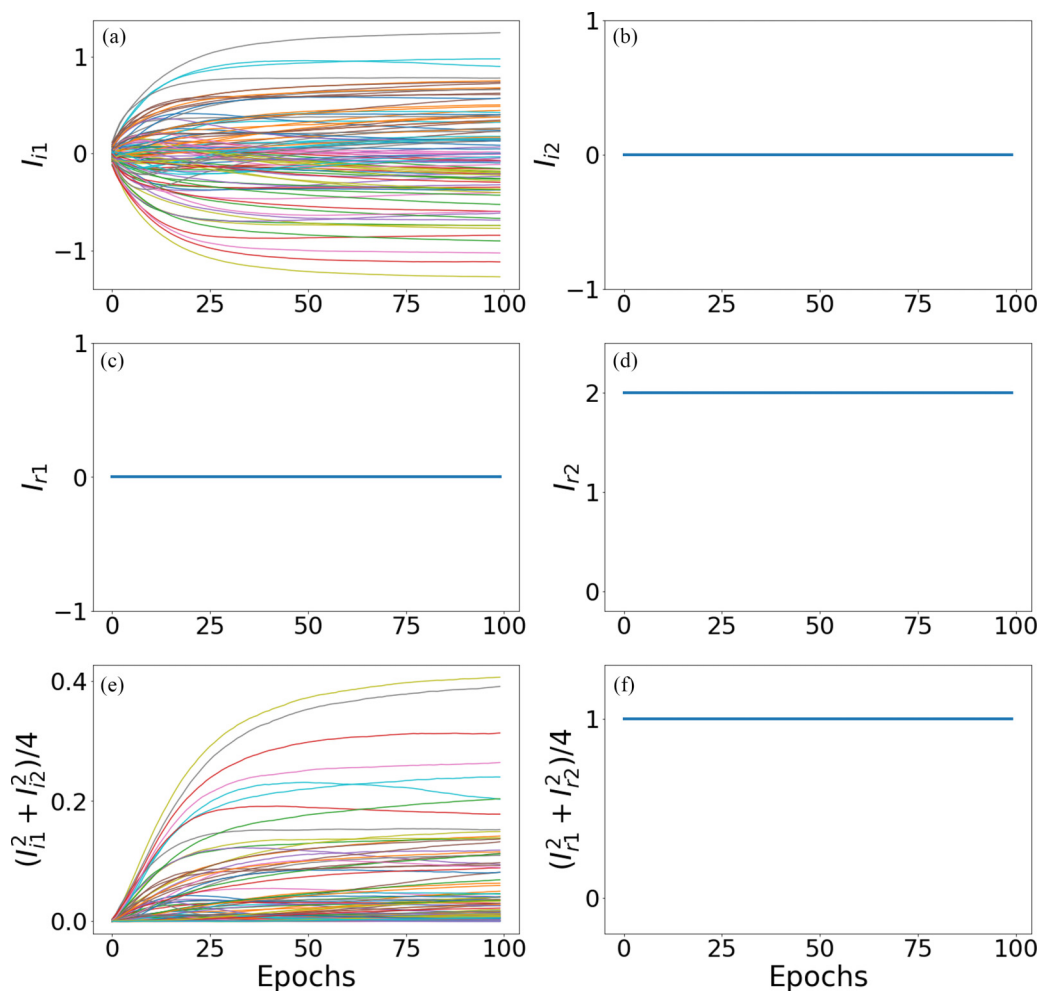


FIG. 8. (a) The invariant  $I_{i1}$  computed from Eq. (C16) using the imaginary part of the OTOC string [defined in Eq. (6)] as a function of training epoch. This invariant changes as the training vector  $\vec{X}$  varies during training and hence can provide information about the learning landscape (b) The invariant  $I_{i2}$  computed from Eq. (C19) using the imaginary part of the OTOC string [defined in Eq. (6)] as a function of training epoch. (c) Same invariant as in (a) but computed with the real part Eq. (C11) of the OTOC string [defined in Eq. (6)]. This remains constant during training (d) Same invariant as in (c) but computed with the real part Eq. (C14) of the OTOC string [defined in Eq. (6)]. This remains constant during training (e) The compound invariant  $\frac{I_{i1}^2 + I_{i2}^2}{4}$  obtained from (a) and (b) vs training epoch. Such compound invariants are plotted in Fig. 2(b) in main text in  $(C_{i\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau))$  vs  $(\dot{C}_{i\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau))$  space. Since (a), (e) is sensitive to  $\vec{X}$  and hence changes during training it is evident why studying such invariants generated from the imaginary part of OTOC strings can be useful compared to the real part alone. (f) The compound invariant as in (e) but for the real part. It shows no change during the training of the network.

$$(ii) (\dot{C}_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)^2 + C_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)^2)$$

*Proof.* Can be verified through explicit evaluation using the solution  $C_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)$  in Corollary 5.1, or by combining the invariants  $I_{r1}, I_{r2}$  in Eqs. (C11) and (C14) as  $\frac{1}{4}(I_{r1}^2 + I_{r2}^2)$ . This invariant is plotted in Fig. 2(a) in the main text for the real and imaginary part. The profile for the invariant is different in two cases with the imaginary part being sensitive to the training process of the network  $G$  unlike the real part and hence can be used to deliver meaningful insight about the learning dynamics. ■

Exactly similar invariants can be obtained for  $C_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)$  by substituting  $C_{r\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau) \rightarrow C_{i\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau)$  in the above expressions. Note that we also display the real-time behavior of the real and imaginary part in Fig. 7.

### 1. Profile of invariants during training

For demonstration as how these invariants of motion changes during training of the network  $G$  to learn the ground state of a given driver in epoch time, we have used the TFIM model [see Eqs. (33) and (34)] as the driver with  $N = 10$  spins. We have discussed the training algorithm in Appendix D. We use  $n = m = N = 10$  spins in the network  $G$  and plot the primitive invariants  $I_{r1}, I_{r2}, I_{i1}, I_{i2}$ , which are discussed and proven in Theorem II.1 in main text and proven in this section before in Theorem C.2 (2). We also plot the compound invariants  $\frac{I_{i2}^2 + I_{i1}^2}{4}$  for each  $k, m$  pair where  $k$  belongs to visible neurons and  $m$  to hidden neurons and  $\frac{I_{r2}^2 + I_{r1}^2}{4}$ , which is displayed in Figs. 2(a) and 2(b) and proven in this section in Corollary 5.2. We use 100 epochs for comparison of all invariants. For a given epoch with the incumbent instance of the parameter vector  $\vec{X}$  one can plot each invariant in the space of

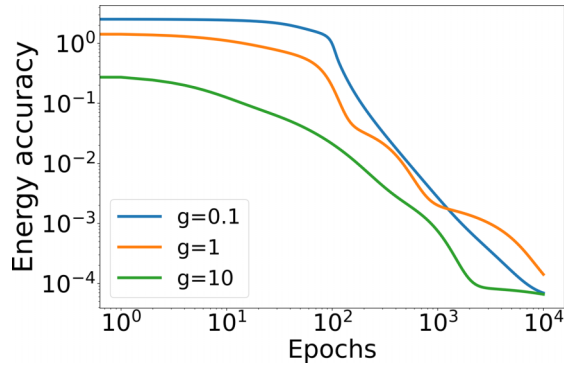


FIG. 9. TFIM.

$\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau) - \dot{\xi}_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau)$  space where  $\xi_{\sigma^\alpha, \sigma^\beta}(\vec{X}, \tau)$  can either be the real or the imaginary part of  $(C_{\sigma^\alpha, \sigma^\beta}(0, 0, \vec{X}, \tau))$  as has been done in the main text. We see that certain invariants computed using the imaginary parts of OTOC string [see Eq. (6) in main text] have nontrivial evolution [for example, see Figs. 8(a) and 8(e)] unlike those computed using the real parts. This gives us a direct evidence why imaginary part of OTOC strings can be of use to gain insight into the learning mechanism, which may not be obtainable from the real part—a claim central to the thesis of our paper. One functional importance that may stem from analyzing such invariants using the imaginary part is identifying certain pair of neurons  $(k, m)$ , which reports an invariant value that remains nearly conserved and close to 0 during the entire course of training. Such neurons can be considered to not undergo information exchange (and hence remains nearly uncorrelated). A common set of neurons of the hidden node, which shares such a property with any of the neurons of the visible set may be considered redundant neurons and hence can be discarded for more compact subsequent trial, which can reduce the cost of the training. Designing markers through which identification of such markers can be enabled through estimation of the invariants may be a fruitful future direction that can benefit from a thorough investigation.

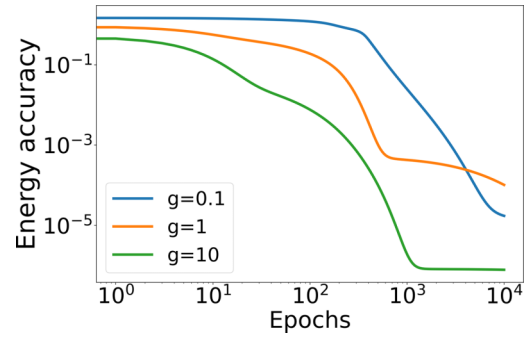


FIG. 10. c-TFIM.

#### APPENDIX D: TRAINING OF TFIM AND c-TFIM AND DEDUCTION OF EIGENVALUES OF ${}^2\rho(v_k, h_m)$

The Figs. 9 and 10 depict the training of RBM network  $G$  for TFIM and c-TFIM model respectively. It shows the variation of Energy accuracy with epochs for training  $G$ . For each training process parameters of RBM are initialized randomly. For training the network we use Variational Monte Carlo technique with Stochastic Reconfiguration based gradient updates as illustrated in Ref. [73] of main text. Learning rate used is 0.05 and  $(n, p) = (4, 4)$  in the network  $G$ . The convergence threshold set is  $\leq 10^{-2}$ . The relative error in the converged state is less than 0.1%. Energy accuracy =  $\langle \psi(\vec{X}) | H | \psi(\vec{X}) \rangle - \lambda_0$  where Hamiltonian  $H$  is that of the driver (TFIM or c-TFIM) and the state  $\psi(\vec{X})$  is the ansatz for the corresponding ground state.  $\lambda_0$  is the smallest eigenvalue (true ground state energy) of the Hamiltonian.

For Figs. 3(d), 3(e), 3(g), and 3(h) in the main text, to obtain each point, we average over all pairs of visible and hidden indices for  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$  (i.e.,  $k = \{1, 2, 3, 4\}$ ,  $m = \{1, 2, 3, 4\}$  implying 16 pairs) and another averaging over a sample of 100 best-converged points obtained from the training of 1000 randomly initialized networks.

#### 1. Deduction of the eigenvalues of ${}^2\rho(v_k, h_m)$

In this subsection we explicitly deduce the eigenvalues of  ${}^2\rho(v_k, h_m)$  in Eqs. (24)–(27) in the main text. To do that let us compute a specific matrix element [say  $(v_k, h_m, v'_k, h'_m)$ th element] of  ${}^2\rho(v_k, h_m)$  by direct contraction of  $\rho_{ih}(\vec{X}, \vec{v}, \vec{h})$  in Eq. (3) as follows:

$$\begin{aligned}
 {}^2\rho(\sigma^z(v_k), \sigma^z(h_m))_{v'_k, h'_m}^{v_k, h_p} &= \frac{\delta_{v_k, v'_k} \delta_{h_m, h'_m}}{\text{Tr}_{\{v, h\}} e^{-\mathcal{H}(\vec{X}, \vec{v}, \vec{h})}} \sum_{v_i \neq v_k} e^{-\beta \sum_i a_i v_i} \sum_{h_j \neq h_m} e^{-\beta \sum_j b_j h_j - \beta \sum_{ij}^{n,p} W_{ij}^i v_i h_j} \\
 &= \frac{\delta_{v_k, v'_k} \delta_{h_m, h'_m}}{\text{Tr}_{\{v, h\}} e^{-\mathcal{H}(\vec{X}, \vec{v}, \vec{h})}} \sum_{v_i \neq v_k} e^{-\sum_i a_i v_i} e^{-b_m h_m - \sum_i W_m^i v_i h_m} \sum_{h_j \neq h_m} \prod_{j \neq m}^p e^{-b_j h_j - \sum_i W_j^i v_i h_j} \\
 &= \frac{\delta_{v_k, v'_k} \delta_{h_m, h'_m}}{\mathcal{Z}} \sum_{v_i \neq v_k} e^{-a_k v_k - b_m h_m - W_m^k v_k h_m} e^{-\sum_{i \neq k} a_i v_i - \sum_i W_m^i v_i h_m} \sum_{h_j \neq h_m} \prod_{j \neq m}^p e^{-b_j h_j - \sum_i W_j^i v_i h_j} \\
 &= \frac{\delta_{v_k, v'_k} \delta_{h_m, h'_m} e^{-a_k v_k - b_m h_m - W_m^k v_k h_m}}{\mathcal{Z}} \sum_{v_i \neq v_k} e^{-\sum_{i \neq k} a_i v_i - \sum_i W_m^i v_i h_m} \sum_{h_j \neq h_m} \prod_{j \neq m}^p e^{-b_j h_j - \sum_i W_j^i v_i h_j}
 \end{aligned}$$

$$\begin{aligned}
&= \frac{\delta_{v_k, v'_k} \delta_{h_m, h'_m} e^{-a_k v_k - b_m h_m - W_m^k v_k h_m}}{Z} \sum_{v_i \neq v_k} e^{-\sum_{i \neq k}^n a_i v_i - W_m^i v_i h_m} \prod_{j \neq m}^p 2 \cosh \left( -b_j - \sum_i^n W_j^i v_i \right) \\
&= \frac{\delta_{v_k, v'_k} \delta_{h_m, h'_m} N_{h_m} e^{-a_k v_k - b_m h_m - W_m^k v_k h_m}}{Z} \left\langle \prod_{j \neq m}^p 2 \cosh \left( b_j + \sum_i^n W_j^i v_i \right) \right\rangle_{P(\{v_i\}_{i \neq k}^n, h_m)}, \quad (D1)
\end{aligned}$$

where in Eq. (D1) the following definitions are used

$$\cosh(x) = \cosh(-x) \quad (D2)$$

$$P(\{v_i\}_{i \neq k}^n, h_m) = \frac{e^{-(\sum_{i \neq k}^n a_i v_i + W_m^i v_i h_m)}}{N_{h_m}} \quad (D3)$$

$$N_{h_m} = \prod_{i \neq k}^n 2 \cosh \left( a_i + \sum_{i \neq k}^n W_m^i h_m \right) \quad (D4)$$

$$Z = \text{Tr}_{\{v, h\}} e^{-\mathcal{H}(\vec{v}, \vec{h})} = \sum_{(\vec{v}, \vec{h})} e^{(-\sum_i^n -a_i v_i - \sum_j^p b_j h_j - \sum_{i,j}^{n,p} W_j^i v_i h_j)} \quad (D5)$$

Equation (D1) indicates due to the presence of  $\delta_{v_k, v'_k} \delta_{h_m, h'_m}$  factors that  ${}^2\rho(v_k, h_m)$  is an entirely diagonal matrix. Thus the 4 eigenvalues of  ${}^2\rho(v_k, h_m)$  can be written as

$$\begin{aligned}
\lambda_i({}^2\rho(v_k, h_m)) &= \lambda({}^2\rho(v_k = x, h_m = y)) \\
&= \frac{N_y}{Z} e^{-a_k x - b_m y - W_m^k x y} \left\langle \prod_{j \neq m}^p 2 \cosh \left( b_j + \sum_{i \neq k}^n W_j^i v_i + W_j^k x \right) \right\rangle_{P(\{v_i\}_{i \neq k}^n, h_m = y)} \quad \forall (x, y) \in \{-1, 1\}^2. \quad (D6)
\end{aligned}$$

Equation (D6) yields the four eigenvalues displayed in Eqs. (24)–(27) for various values of  $(x, y)$ .

#### APPENDIX E: RELATIONSHIP BETWEEN $\eta(\vec{X})$ [SEE EQ. (9) IN MAIN TEXT] AND $\mathcal{I}(v_k, h_m)$ —PROOF OF THEOREM 2 IN MAIN TEXT

The four eigenvalues  $\{\lambda_i({}^2\rho(v_k, h_m))\}_{i=1}^4$  of the two-particle density matrix  ${}^2\rho(v_k, h_m)$  for the learner  $G$  between a specific pair of visible and hidden spins [say  $(k, m)$ ] as deduced in the previous section can be readily expressed as

$$\lambda_1({}^2\rho(v_k, h_m)) = \lambda({}^2\rho(v_k = 1, h_m = 1)) = \frac{N_1}{Z} e^{-a_k - b_m - W_m^k} \left\langle \prod_{j \neq m}^p 2 \cosh \left( b_j + \sum_{i \neq k}^n W_j^i v_i + W_j^k \right) \right\rangle_{P(\{v_i\}_{i \neq k}^n, h_m = 1)}, \quad (E1)$$

$$\lambda_2({}^2\rho(v_k, h_m)) = \lambda({}^2\rho(v_k = 1, h_m = -1)) = \frac{N_{-1}}{Z} e^{-a_k + b_m + W_m^k} \left\langle \prod_{j \neq m}^p 2 \cosh \left( b_j + \sum_{i \neq k}^n W_j^i v_i + W_j^k \right) \right\rangle_{P(\{v_i\}_{i \neq k}^n, h_m = -1)}, \quad (E2)$$

$$\lambda_3({}^2\rho(v_k, h_m)) = \lambda({}^2\rho(v_k = -1, h_m = 1)) = \frac{N_1}{Z} e^{a_k - b_m + W_m^k} \left\langle \prod_{j \neq m}^p 2 \cosh \left( b_j + \sum_{i \neq k}^n W_j^i v_i - W_j^k \right) \right\rangle_{P(\{v_i\}_{i \neq k}^n, h_m = 1)}, \quad (E3)$$

$$\lambda_4({}^2\rho(v_k, h_m)) = \lambda({}^2\rho(v_k = -1, h_m = -1)) = \frac{N_{-1}}{Z} e^{a_k + b_m - W_m^k} \left\langle \prod_{j \neq m}^p 2 \cosh \left( b_j + \sum_{i \neq k}^n W_j^i v_i - W_j^k \right) \right\rangle_{P(\{v_i\}_{i \neq k}^n, h_m = -1)}, \quad (E4)$$

where each of the averages are computed over the distribution  $P(\{v_i\}_{i \neq k}^n, h_m)$  and  $N_{h_m}$  is the associated normalization constant. These are defined as

$$P(\{v_i\}_{i \neq k}^n, h_m) = \frac{e^{-(\sum_{i \neq k}^n a_i v_i + W_m^i v_i h_m)}}{N_{h_m}}, \quad (E5)$$

$$N_{h_m} = \prod_{i \neq k}^n 2 \cosh \left( a_i + \sum_{i \neq k}^n W_m^i h_m \right). \quad (E6)$$

The corresponding eigenvectors of the two particle density matrix for the eigenvalues in Eqs. (E1)–(E4) are  $|0(v_k)0(h_m)\rangle$ ,  $|0(v_k)1(h_m)\rangle$  and  $|1(v_k)0(h_m)\rangle$ ,  $|1(v_k)1(h_m)\rangle$  respectively for the four eigenvalues Eqs. (E1)–(E4) where  $(0,1)$  is notationally equivalent to  $(1, -1)$  for each spins  $(v_k, h_m)$ .

The quantity  $Z$  is the partition function defined as

$$Z = \sum_{(\vec{v}, \vec{h})} e^{(-\sum_i^n -a_i v_i - \sum_i^p b_j h_j - \sum_{i,j}^{n,p} W_{ij}^1 v_i h_j)}. \tag{E7}$$

However  $Z$  need not be explicitly computed as it can be eliminated using the normalization condition of the eigenvalues. The eigenvalues for one-particle density matrix  ${}^1\rho(\xi_i, 0)$  for a neuron  $\xi_i$  in the learner  $G$ , by contraction from Eqs. (E1)–(E4) are

$$\begin{aligned} \lambda_1({}^1\rho(\xi_i)) &= \lambda_i({}^2\rho(v_k, h_m)) + \lambda_j({}^2\rho(v_k, h_m)) \\ &\quad (\text{if } \xi_i = v_k, \quad (i, j) = (1, 3)) \\ &\quad (\text{if } \xi_i = h_m, \quad (i, j) = (1, 4)), \end{aligned} \tag{E8}$$

$$\begin{aligned} \lambda_2({}^1\rho(\xi_i)) &= \lambda_i({}^2\rho(v_k, h_m)) + \lambda_j({}^2\rho(v_k, h_m)) \\ &\quad (\text{if } \xi_i = v_k, \quad (i, j) = (2, 4)) \\ &\quad \text{if } (\xi_i = h_m, \quad (i, j) = (2, 3)) \end{aligned} \tag{E9}$$

with respective eigenvectors are  $|0(\xi_i)\rangle$  and  $|1(\xi_i)\rangle$  where  $\xi_i \in (v_k, h_m)$ .

Now using these information, one can deduce expressions for  $S({}^2\rho(v_k, h_m))$  and  $S({}^1\rho(v_k))$ ,  $S({}^1\rho(h_m))$  and hence of  $\mathcal{I}(v_k, h_m)$  as follows:

$$\begin{aligned} \mathcal{I}(v_k, h_m) &= S({}^1\rho(v_k)) + S({}^1\rho(h_m)) - S({}^2\rho(v_k, h_m)) \\ &= -\lambda_1({}^1\rho(v_k))\log_2(\lambda_1({}^1\rho(v_k))) - \lambda_2({}^1\rho(v_k))\log_2(\lambda_2({}^1\rho(v_k))) - \lambda_1({}^1\rho(h_m))\log_2(\lambda_1({}^1\rho(h_m))) \\ &\quad - \lambda_2({}^1\rho(h_m))\log_2(\lambda_2({}^1\rho(h_m))) + \lambda_1({}^2\rho(v_k, h_m))\log_2(\lambda_1({}^2\rho(v_k, h_m))) + \lambda_2({}^2\rho(v_k, h_m))\log_2(\lambda_2({}^2\rho(v_k, h_m))) \\ &\quad + \lambda_3({}^2\rho(v_k, h_m))\log_2(\lambda_3({}^2\rho(v_k, h_m))) + \lambda_4({}^2\rho(v_k, h_m))\log_2(\lambda_4({}^2\rho(v_k, h_m))) \\ &\quad - (\lambda_1({}^2\rho(v_k, h_m)) + \lambda_3({}^2\rho(v_k, h_m))\log_2(\lambda_1({}^2\rho(v_k, h_m)) + \lambda_3({}^2\rho(v_k, h_m))) \\ &\quad - (\lambda_2({}^2\rho(v_k, h_m)) + \lambda_4({}^2\rho(v_k, h_m))\log_2(\lambda_2({}^2\rho(v_k, h_m)) + \lambda_4({}^2\rho(v_k, h_m))) \\ &\quad - (\lambda_1({}^2\rho(v_k, h_m)) + \lambda_4({}^2\rho(v_k, h_m))\log_2(\lambda_1({}^2\rho(v_k, h_m)) + \lambda_4({}^2\rho(v_k, h_m))) \\ &\quad - (\lambda_2({}^2\rho(v_k, h_m)) + \lambda_3({}^2\rho(v_k, h_m))\log_2(\lambda_2({}^2\rho(v_k, h_m)) + \lambda_3({}^2\rho(v_k, h_m))) \\ &\quad + \lambda_1({}^2\rho(v_k, h_m))\log_2(\lambda_1({}^2\rho(v_k, h_m))) + \lambda_2({}^2\rho(v_k, h_m))\log_2(\lambda_2({}^2\rho(v_k, h_m))) \\ &\quad + \lambda_3({}^2\rho(v_k, h_m))\log_2(\lambda_3({}^2\rho(v_k, h_m))) + \lambda_4({}^2\rho(v_k, h_m))\log_2(\lambda_4({}^2\rho(v_k, h_m))) \end{aligned} \tag{E10}$$

wherein in the last equality in Eq. (E10), contraction schemes from Eqs. (E8) and (E9) were used. Equation (E10) thus expresses  $\mathcal{I}(v_k, h_m)$  in terms of the eigenvalues of  ${}^2\rho(v_k, h_m)$ . Apart from this the normalization condition also inter-relates the eigenvalues as

$$\lambda_1({}^2\rho(v_k, h_m)) + \lambda_2({}^2\rho(v_k, h_m)) + \lambda_3({}^2\rho(v_k, h_m)) + \lambda_4({}^2\rho(v_k, h_m)) = 1. \tag{E11}$$

The expression for the  $\eta(\vec{X})$  as follows:

$$\begin{aligned} \eta(\vec{X}) &= \text{Cov}(\sigma^z(v_k, 0), \sigma^z(h_m, 0)) = \text{Tr}(\sigma^z(v_k, 0)\sigma^z(h_m, 0) {}^2\rho(v_k, h_m)) - \langle \sigma^z(v_k, 0) \rangle_{\rho_{v_k}} \langle \sigma^z(h_m, 0) \rangle_{\rho_{h_m}} \\ &= \lambda_1({}^2\rho(v_k, h_m)) - \lambda_2({}^2\rho(v_k, h_m)) - \lambda_3({}^2\rho(v_k, h_m)) + \lambda_4({}^2\rho(v_k, h_m)) - \left( \sum_{i=1,3} \lambda_i({}^2\rho(v_k, h_m)) - \sum_{i=2,4} \lambda_i({}^2\rho(v_k, h_m)) \right) \\ &\quad \times \left( \sum_{i=1,4} \lambda_i({}^2\rho(v_k, h_m)) - \sum_{i=2,3} \lambda_i({}^2\rho(v_k, h_m)) \right). \end{aligned} \tag{E12}$$

Also to ensure positive semidefiniteness of  ${}^2\rho(v_k, h_m)$ , we have

$$\lambda_i({}^2\rho(v_k, h_m)) \geq 0 \quad \forall i \in \{1, 2, 3, 4\}. \tag{E13}$$

Thus we see that Eq. (E10) combined with (E11), Eq. (E13) for a given value of  $\eta(\vec{X})$  in Eq. (E12) completely defines the  $\mathcal{I} - \eta$  space in terms of the spectrum of  ${}^2\rho(v_k, h_m)$ .

Lower bound (LB)

The minimization of Eq. (E10) over the spectrum of  ${}^2\rho(v_k, h_m)$  with the constraints defined in Eqs. (E11), (E13), and (E12) [for a fixed value of  $\eta(\vec{X})$ ] gives the following condition:

$$\begin{aligned}\lambda_1({}^2\rho(v_k, h_m)) &= \lambda_2({}^2\rho(v_k, h_m)) = \lambda_a(\text{say}), \\ \lambda_3({}^2\rho(v_k, h_m)) &= \lambda_4({}^2\rho(v_k, h_m)) = \lambda_b(\text{say}).\end{aligned}\quad (\text{E14})$$

Substituting the above conditions in Eq. (E11) and in Eq. (E10) leads to the following:

$$\lambda_b = \frac{1}{2} - \lambda_a, \quad (\text{E15})$$

$$\eta(\vec{X}) = 4\lambda_a - 1 \quad [\because \text{Eq. (E15)}]. \quad (\text{E16})$$

Substituting Eqs. (E16), (E15) and the conditions in Eq. (E14), in Eq. (E10) yields the lower bound  $LB$  as

$$\begin{aligned}LB &= \mathcal{I}_{LB}(v_k, h_m) = 2 + 2\lambda_a \log_2(\lambda_a) + 2\lambda_b \log_2(\lambda_b) \\ &= 2 + 2\lambda_a \log_2(\lambda_a) + 2\left(\frac{1}{2} - \lambda_a\right) \log_2\left(\frac{1}{2} - \lambda_a\right) \\ &= 2 + 2\lambda_a \log_2(\lambda_a) + (1 - 2\lambda_a) \log_2\left(\frac{1}{2} - \lambda_a\right) \\ &= 2 + \left(\frac{\eta(\vec{X}) + 1}{2}\right) \log_2\left(\frac{\eta(\vec{X}) + 1}{4}\right) + \left(\frac{1 - \eta(\vec{X})}{2}\right) \log_2\left(\frac{1 - \eta(\vec{X})}{4}\right) \quad \because \text{Eq. (E16)}.\end{aligned}\quad (\text{E17})$$

Note that  $LB$  is symmetric about  $\eta(\vec{X}) = 0$  and remains unchanged on substitution of  $\eta(\vec{X}) \rightarrow -\eta(\vec{X})$ . Also,  $LB$  is mathematically only defined if  $-1 \leq \eta(\vec{X}) \leq 1$ , which is true for covariances of Pauli operators in the learner  $G$  (a direct byproduct of Cauchy-Schwartz inequality and idempotency of Pauli operators, which bounds their respective variances to within 1).

Upper bound (UB)

The maximization of Eq. (E10) over the spectrum of  ${}^2\rho(v_k, h_m)$  with the constraints defined in Eqs. (E11), (E13), and (E12) [for a fixed value of  $\eta(\vec{X})$ ] gives the following condition:

$$\lambda_1({}^2\rho(v_k, h_m)) = \lambda_2({}^2\rho(v_k, h_m)) = 0, \quad \lambda_3({}^2\rho(v_k, h_m)) = \lambda_a(\text{say}), \quad \lambda_4({}^2\rho(v_k, h_m)) = \lambda_b(\text{say}). \quad (\text{E18})$$

Now using Eq. (E18) in (E11) and (E12) we get

$$\lambda_b = 1 - \lambda_a, \quad (\text{E19})$$

$$\eta(\vec{X}) = -1 + (2\lambda_a - 1)^2 = 4\lambda_a(4\lambda_a - 1) \quad [\because \text{Eq. (E19)}],$$

$$\lambda_a = \frac{1 \pm \sqrt{1 + (-1)^\gamma \eta(\vec{X})}}{2}, \quad (\text{E20})$$

wherein in the last equality to maintain positive semidefiniteness of  $\lambda_a$  (hence for  $\lambda_b$ ) in both the roots, a factor of  $(-1)^\gamma$  was used along with the condition  $-1 \leq \eta(\vec{X}) \leq 1$ . Note that  $\gamma = 0$  when  $\eta(\vec{X}) < 0$  and  $\gamma = 1$  when  $\eta(\vec{X}) \geq 0$ . Now substituting Eqs. (E18), (E19), and (E20) in (E10) we obtain UB as

$$\begin{aligned}UB &= \mathcal{I}_{UB}(v_k, h_m) = -\lambda_a \log_2(\lambda_a) - \lambda_b \log_2(\lambda_b) = -\lambda_a \log_2(\lambda_a) - (1 - \lambda_a) \log_2(1 - \lambda_a) \quad [\because \text{Eq. (E19)}] \\ &= -\left(\frac{1}{2} + \frac{\sqrt{1 + (-1)^\gamma \eta(\vec{X})}}{2}\right) \log_2\left(\frac{1}{2} + \frac{\sqrt{1 + (-1)^\gamma \eta(\vec{X})}}{2}\right) - \left(\frac{1}{2} - \frac{\sqrt{1 + (-1)^\gamma \eta(\vec{X})}}{2}\right) \log_2\left(\frac{1}{2} - \frac{\sqrt{1 + (-1)^\gamma \eta(\vec{X})}}{2}\right)\end{aligned}\quad (\text{E21})$$

$$[\because \text{Eq. (E20)}] \quad (\text{E22})$$

where substitution of either root from Eq. (E20) would lead to the same  $UB$  due to symmetry.

#### APPENDIX F: SATURATION OF LOWER BOUND (LB) IN $\mathcal{I}(v_k, h_m)$ AND $\eta(\vec{X})$ SPACE IN EIGENPAIR LEARNING OF NETWORK $G$

In this section we discuss the several systems we have used as a driver for our task of learning eigenpairs and training the network  $G = (V, E)$ . We use for demonstration a wide variety of systems wherein the ground state is non-negative

due to Perron-Frobenius theorem as discussed in the main text. Each of these model is endowed with a hamiltonian  $H(\lambda_1, \lambda_2, \lambda_3 \dots \lambda_n)$  with several generic controllable parameters  $\{\lambda_i\}_{i=1}^n$ . Tuning these controllable parameters allows one to access ground states with various phase properties for different sizes of the system. We show that for each cases for different sizes, both the assertions in the text—(a) satu-

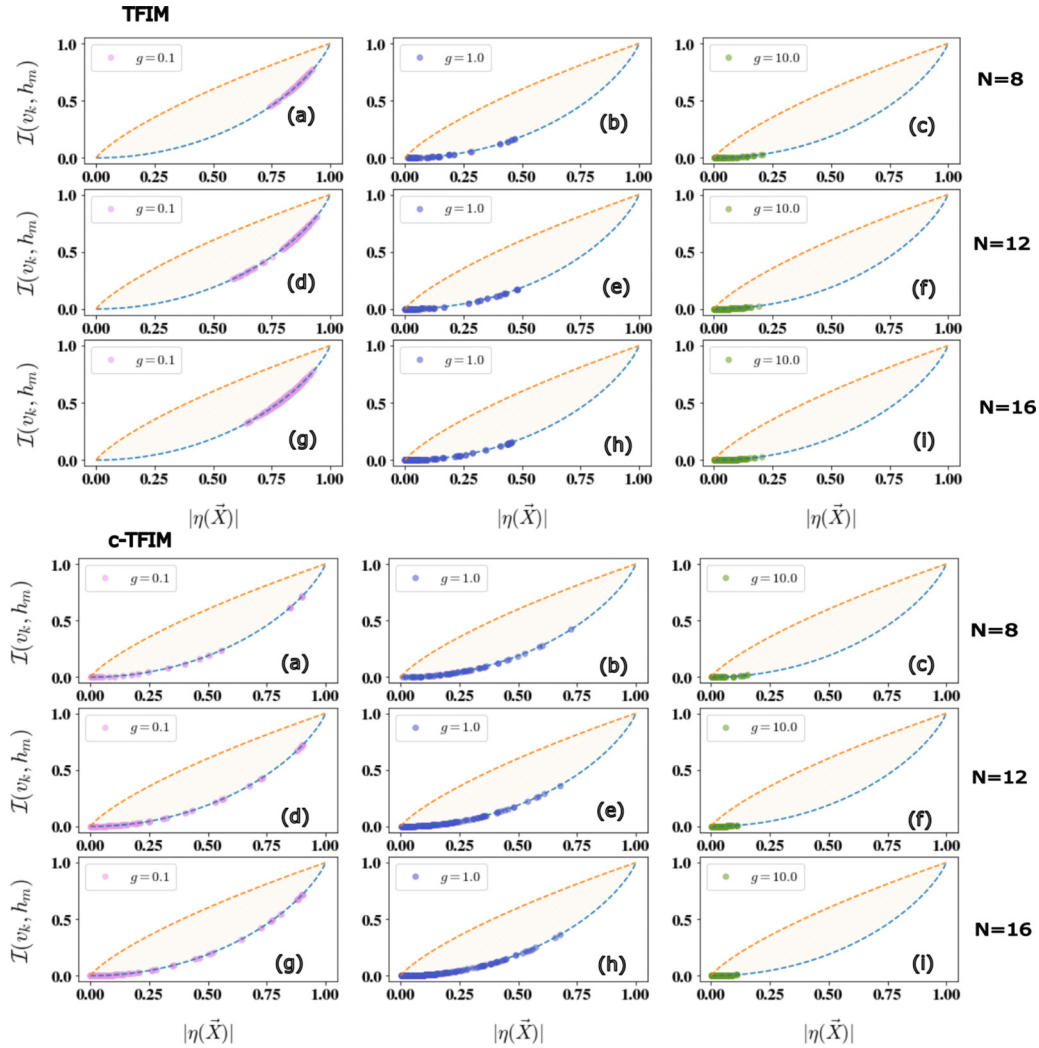


FIG. 11. (a) The saturation of the lower bound for  $g = 0.1$ . (b) Same as in (a) for  $g = 1.0$  (c) for  $g = 10.0$  and  $N = 8$ . The plots also show clustering of  $(|\eta(\vec{X})|, \mathcal{I}(v_k, h_m))$  near upper regime for  $g = 0.1$ , middle and lower regime in  $g = 1.0$  and close to the origin in  $g = 10.0$  (d)–(f) Same as in (a–c) but for  $N = 12$ . (g)–(i) Same as in (a)–(c) but for  $N = 16$ . For the c-TFIM panel we have similar plots as in TFIM (a)–(i). Note that the spread of the points in c-TFIM is larger than in TFIM, feature consistent with the discussion in the main text, which says many equivalent representations the network chooses for volume-law connectivity.

ration of lower bound (LB) in the  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$  space by the representation chosen by the trained/learned network, (b) sliding of the  $(|\eta(\vec{X})|, \mathcal{I}(v_k, h_m))$  points on the LB as the controllable parameters  $\{\lambda\}_{i=1}^n$  are tuned across the phase boundaries. The last point evidentiates how spin correlation among the sub-units or spins of the actual driver system gets translated or mimicked on the spin correlation between the visible and hidden sub-units of the learner network  $G$ . It must be emphasized this associates a functionally quantifiable importance of the neurons of the hidden layer, which are oblivious to the spins of the driver. It is only the spins of the visible neurons whose state in the basis of its acceptable configurations is trained to represent the physical ground state of the driver.

We start with the familiar TFIM and c-TFIM, which has been explicitly discussed in the main text. We have shown in Fig. 4 that for  $N = 4, 6, 10, 20$  for both TFIM and c-TFIM the lower bound (LB) in the  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$  space is saturated and in Fig. 5 we show that mean  $\mathcal{I}(v_k, h_m)$  and  $|\eta(\vec{X})|$  values

slides along the LB as the controllable parameter  $g$  in  $H(g)$  is changed from a ferromagnet to a disordered phase for sizes  $N = 4, 6, 8, 10, 12, 14, 16, 18, 20, 24$ . Herein we show that the same assertions hold for all intermediate sizes as well for both the models TFIM and c-TFIM with  $n = p = N$  for the network  $G$ . The results are displayed in Fig. 11

We next discuss the Sherrington-Kirkpatrick (SK) model with transverse magnetic field. The form of the Hamiltonian is

$$H = -B \sum_{i_d} \sigma^x(i_d) - \sum_{i_d j_d} J_{i_d j_d} \sigma^z(i_d) \sigma^z(j_d) \quad (\text{F1})$$

where unlike in TFIM and c-TFIM, each of the the coupling matrix elements  $J_{i_d j_d}$  are different and is randomly sampled from a normal distribution, i.e.,  $\mathcal{N}(0, 1)$ . Note that this can mean a model with extreme inhomogeneous couplings too such that interaction strength along the length of the sites

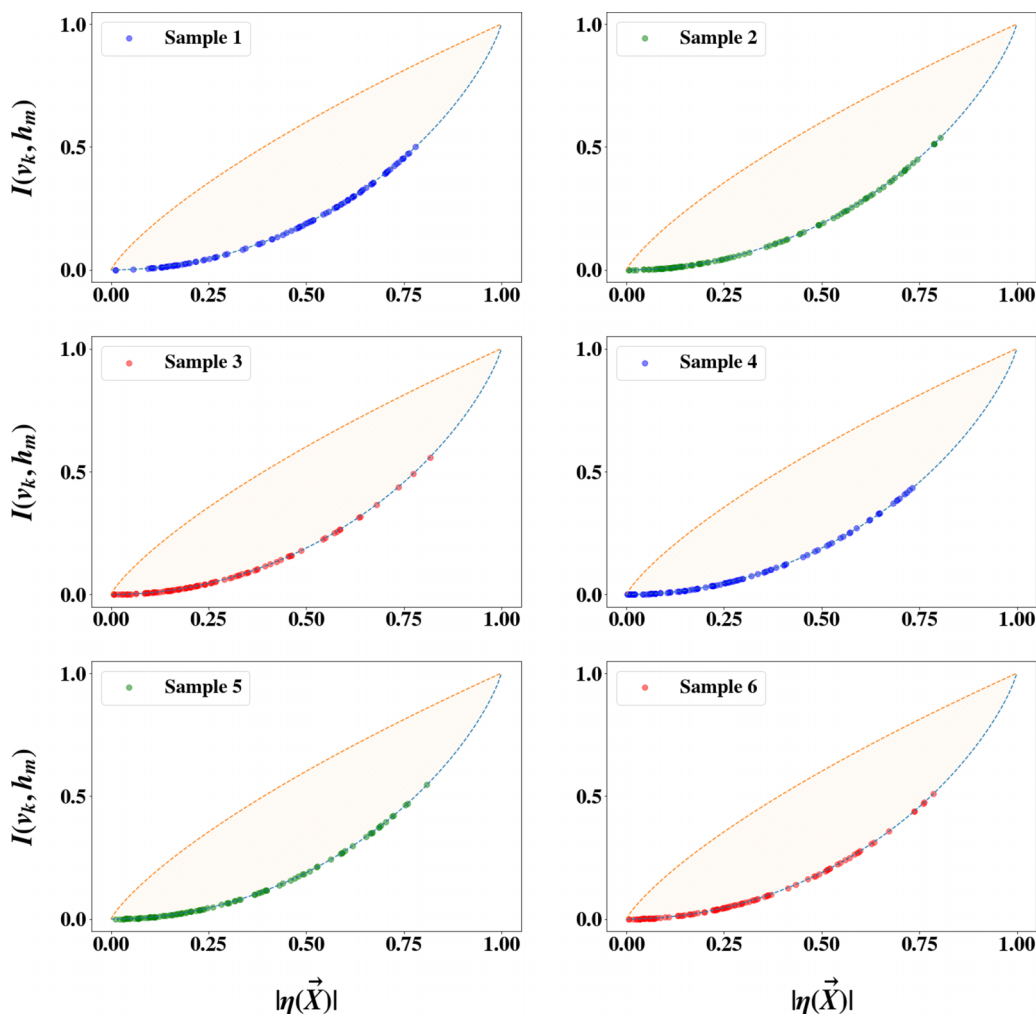


FIG. 12. The saturation of the lower bound in the  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$  space for all five samples studied of the SK model. The sample indices are displayed in the corresponding legend.

do not change monotonically unlike the connectivity pattern used in c-TFIM and TFIM which. We draw five samples from that distribution and for each train the model to show the representation chosen by the network for the ground state and display the results in Fig. 12. In each case we see the  $(|\eta(\vec{X})|, \mathcal{I}(v_k, h_m))$  pair saturates the lower bound. Note that there is no analogous definition of  $g$  in this case as has been defined for TFIM and c-TFIM as the couplings are inhomogeneous. We use  $N = n = p = 10$  and  $B = 1$  for all calculations.

We now turn our attention to another spin model that unlike the previous set of models also have another set of interaction terms involving  $\sigma^y(i_d)\sigma^y(j_d)$ . Moreover the interactions are even anisotropic. The Hamiltonian of the system is

$$H = -B \sum_{i_d} \sigma^x(i_d) - \sum_{\langle i_d j_d \rangle} J(1 + \gamma) \sigma^z(i_d) \sigma^z(j_d) - \sum_{\langle i_d j_d \rangle} J(1 - \gamma) \sigma^y(i_d) \sigma^y(j_d). \quad (\text{F2})$$

The system possesses ground state with non-negative coefficients too as a consequence of Perron-Frobenius theorem.

We demonstrate even for this system with anisotropic YZ type interaction profile the representation chosen by the trained state of the learner ground state saturates the lower bound in the  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$  space. We choose  $n = p = N = 10$  spins and vary  $(g, \gamma)$  in Eq. (F2) as shown in Fig. 13. The parameter  $g = B/J$  is defined as in the case of TFIM and c-TFIM.

#### APPENDIX G: EFFECT OF HIDDEN NODE DENSITY ON TRAINING OF NETWORK $G$

In this section we provide a systematic study of the variation in hidden node density for the network  $G$  while training the latter network for obtaining the ground state of TFIM model [see Eqs. (33) and (34) in main text]. The hidden node density is defined as the ratio of the number of hidden neurons  $p$  vs the number of visible  $n$  neurons used in the network  $G = (V, E)$ , i.e.,  $\alpha = \frac{p}{n}$ . We use for demonstration the said TFIM model as the driver with  $N = 10$  spins, which corresponds to a Hilbert space of dimension  $2^{20}$ . We vary the  $g = [0.2, 1.0, 5.0]$  parameter of the driver to study the ground states in various phases (see main text for details). The network  $G$  is trained with  $n = N = 10$  spins in the visible layer



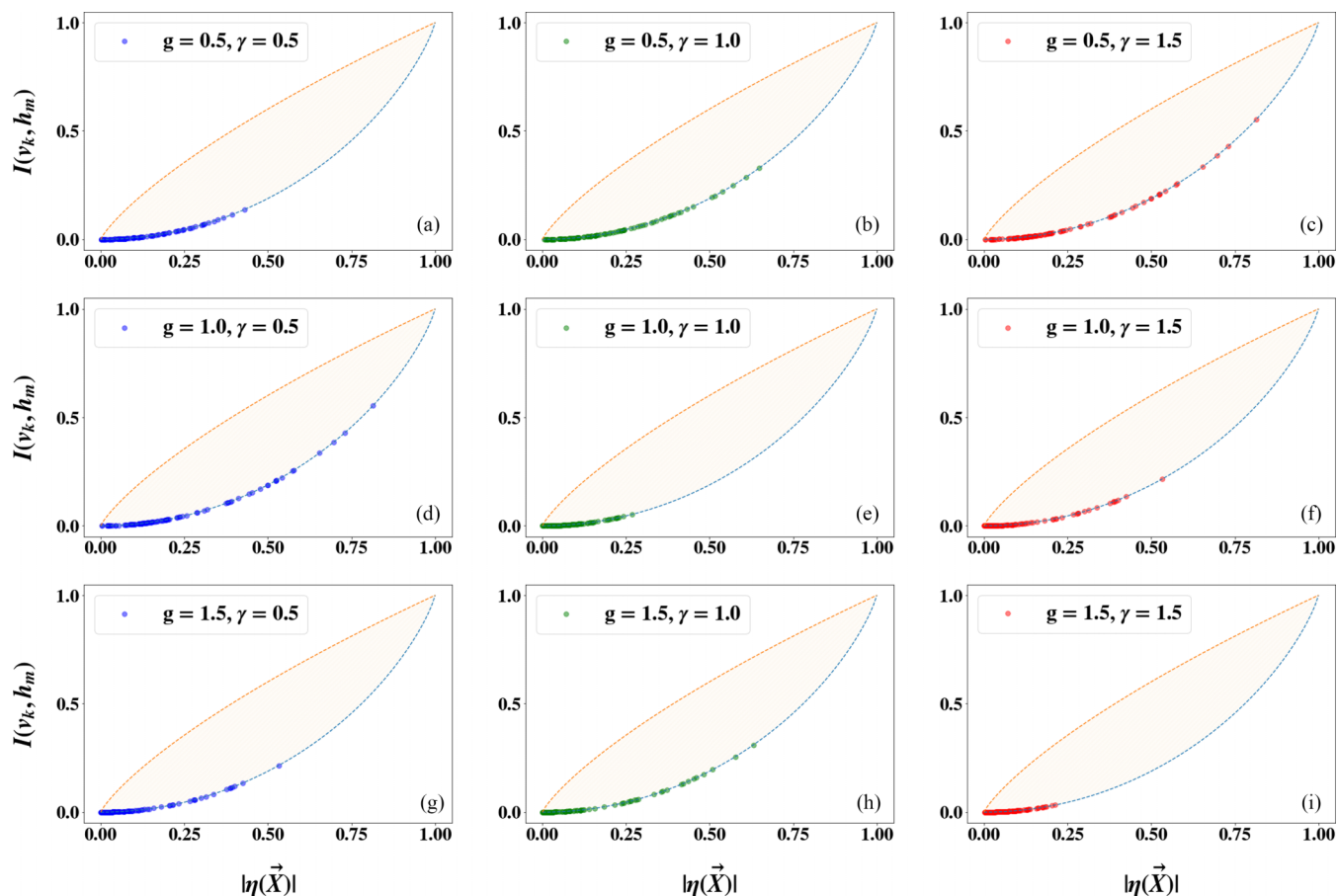


FIG. 13. The saturation of the lower bound in the  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$  space for the trained learner for the ground state of Eq. (F2) corresponding to (a)  $(g, \gamma) = (0.5, 0.5)$ , (b)  $(g, \gamma) = (0.5, 1.0)$ , (c)  $(g, \gamma) = (0.5, 1.5)$ , (d)  $(g, \gamma) = (1.0, 0.5)$ , (e)  $(g, \gamma) = (1.0, 1.0)$ , (f)  $(g, \gamma) = (1.0, 1.5)$ , (g)  $(g, \gamma) = (1.5, 0.5)$ , (h)  $(g, \gamma) = (1.5, 1.0)$ , (i)  $(g, \gamma) = (1.5, 1.5)$ .

and  $p = \alpha n$  spins in the hidden layer wherein  $\alpha \in [1, 2, 3]$ . The results are displayed in the Fig. 14. In Figs. 14(a)–14(c) shows the accuracy threshold reached compared to exact diagonalization when the network is trained using different  $\alpha$  for various values of  $g$ . We see in all cases the acquired energy error in the trained network is  $10^{-2} - 10^{-4}$ , which registers a relative error percentage of  $\leq 0.1\%$  in the worst case with no appreciable dependence on  $\alpha$  within the range studied.

We use  $\alpha = 1$  for all results in the main text and in the remaining portion of Appendix unless otherwise specified. In Figs. 14(d)–14(f) we see that the final trained state is an eigenstate of the Hamiltonian with a very low energy variance of  $\leq 10^{-2}$ . In Figs. 14(g) and 14(h) we plot the representations chosen by the trained network in the  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$  space discussed in main text. We see both our assertions are individually valid, i.e., remain true even when  $\alpha$  is varied. Firstly we see the trained state for all values of  $\alpha$  saturates the lower bound and secondly the mean density of points shifts towards the origin in the  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$  space signalling that the trained state of the learner  $G$  is devoid of spin correlation between the hidden and the visible units mirroring the same behavior between the spins of the driver when  $g \rightarrow \infty$  even though the hidden units are not directly related to the driver spins.

#### APPENDIX H: STANDARD DEVIATION ASSOCIATED WITH THE MEAN $\mathcal{I}(v_k, h_m)$ AND MEAN $\eta(\vec{X})$ FOR TFIM AND c-TFIM

In this section we display the standard deviation ( $\sqrt{\text{Var}(X)}$ ) where  $X$  can be the mean values of variates [like  $\mathcal{I}(v_k, h_m)$  etc.] obtained from the mean  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$  values displayed in Figs. 5(a)–5(d).  $\text{Var}(\cdot)$  is the variance of the respective variate. We see from the standard deviation plot below that the quantity for c-TFIM is nearly 1.5–2.0 times higher than that of TFIM for a given  $g$  and a given size  $N$  especially in the low  $g$  limit ( $g \rightarrow 0_+$ ). The results are displayed in Fig. 15.

#### APPENDIX I: VON-NEUMANN ENTROPY OF TFIM AND c-TFIM

The von Neumann entropy of the ground state for (a) TFIM (Fig. 16) and (b) c-TFIM (Fig. 17) models across the central cut (passing through the mid-point of the chain) for varying system sizes ( $N$ ) is depicted. The ground state of the TFIM model obeys the area-law (and hence is constant in 1D) while that of c-TFIM follows the Volume-Law scaling of entanglement entropy and increases proportionally to the increasing size of the chain.

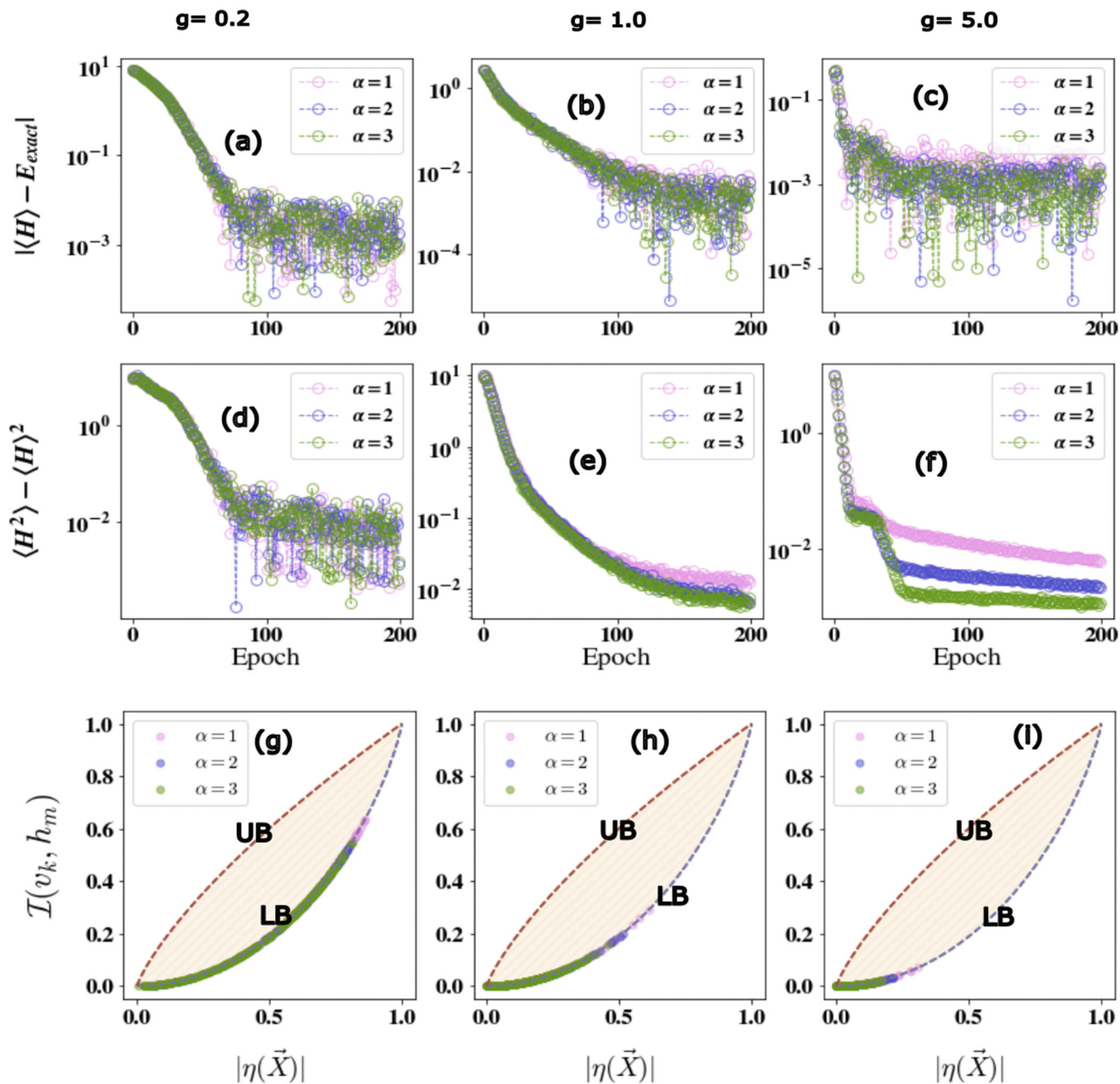


FIG. 14. The energy error (with respect to exact diagonalization) acquired for training TFIM model with  $n = N = 10$ ,  $p = \alpha n$  and  $\alpha = [1, 2, 3]$  for (a)  $g = 0.2$ , (b)  $g = 1$  (critical point), and (c)  $g = 5$ . The resultant energy variance of the trained state with  $\alpha = [1, 2, 3]$  for (a)  $g = 0.2$ , (b)  $g = 1$  (critical point), and (c)  $g = 5$ . The representation of the trained state in  $\mathcal{I}(v_k, h_m)$  and  $\eta(\vec{X})$  space for (a)  $g = 0.2$ , (b)  $g = 1$  (critical point), and (c)  $g = 5$  showing saturation of lower bound in all cases and progressive mirroring of the correlation induced between the visible and hidden spins due to the spin correlation within the driver model.

**APPENDIX J: FISHER INFORMATION OF TFIM AND c-TFIM**

The variation of the largest eigenvalue of the Fisher information matrix with  $g$  for (a) TFIM and (b) c-TFIM models.

The color gradation depicts the increasing value of  $g$  from red to blue. Each point on the Figs. 18 and 19 is obtained by averaging over several best-converged points after training many randomly initialized networks.

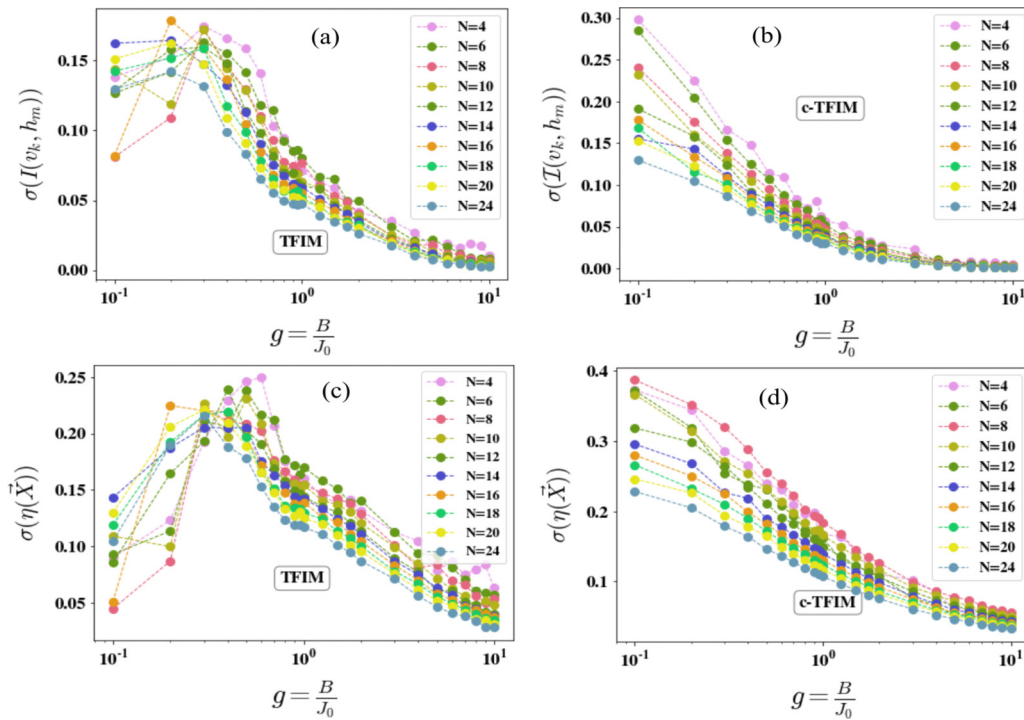


FIG. 15. (a) The standard deviation associated with the averaging process of  $\mathcal{I}(v_k, h_m)$  [where the mean is obtained by averaging over all  $mn$  ( $k, m$ ) pairs of visible and hidden neurons and different initializations; see Sec. IV] for TFIM for all sizes of the driver. (c) The standard deviation associated with the averaging process of the mean  $\eta(\bar{X})$  [where the mean is obtained by averaging over all  $mn$  ( $k, m$ ) pairs of visible and hidden neurons and different initializations; see Sec. IV] for TFIM for all sizes of the driver. [(b),(d)] Same as in [(a),(c)] but for c-TFIM. It is clear from the standard deviation plots that the variance of the  $\mathcal{I}(v_k, h_m)$  and  $\eta(\bar{X})$  is much larger by almost 1.5–2.0 times (especially in the low  $g$  limit) in c-TFIM [see (b),(d)] than in TFIM [see (a),(c)] for all values of  $N$ . Since the variance is computed over all ( $k, m$ ) pairs and over many initialization, this indicates many compatible configurations/representations for a given  $N$  that the trained network can “learn” and display for the volume-law entangled c-TFIM than the area-law entangled TFIM. In the higher  $g$  limit unanimity sets in both models as both the drivers display an unentangled ground state devoid of spin correlation. (See Sec. IV.)

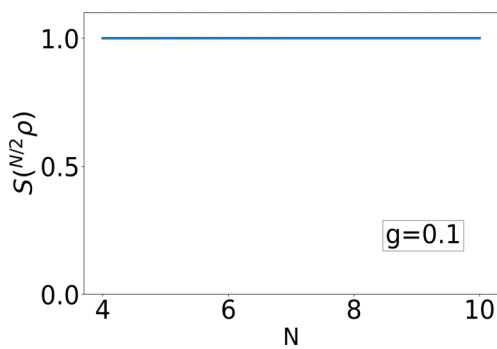


FIG. 16. TFIM.

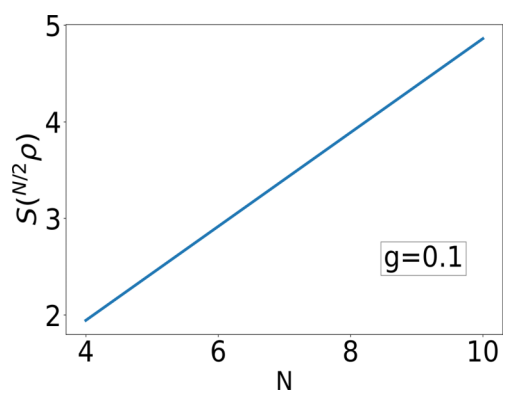


FIG. 17. c-TFIM.

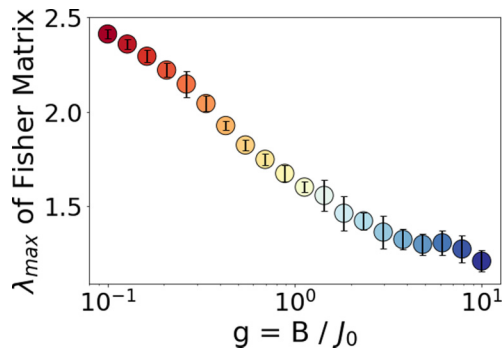


FIG. 18. TFIM.

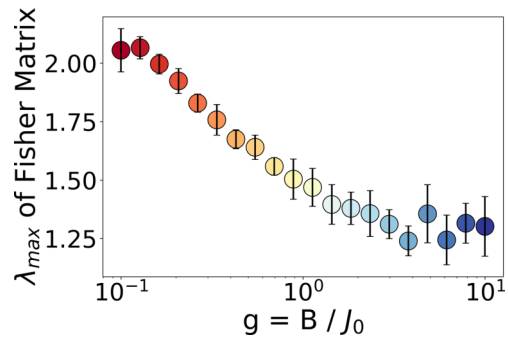


FIG. 19. c-TFIM.

- [1] A. Burkov, *The Hundred-Page Machine Learning Book* (Andriy Burkov Quebec City, QC, Canada, 2019), Vol. 1.
- [2] I. H. Sarker, Machine learning: Algorithms, real-world applications and research directions, *SN Comput. Sci.* **2**, 160 (2021).
- [3] D. Dhall, R. Kaur, and M. Juneja, Machine learning: A review of the algorithms and its applications, in *Proceedings of ICRIC 2019*, edited by P. K. Singh, A. K. Kar, Y. Singh, M. H. Kolekar, and S. Tanwar (Springer International Publishing, Cham, 2020), pp. 47–63.
- [4] G. A. Tsahrintzis and L. C. Jain, *Machine Learning Paradigms: Advances in Deep Learning-Based Technological Applications* (Springer, New York, 2020), Vol. 18.
- [5] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, Machine learning and the physical sciences, *Rev. Mod. Phys.* **91**, 045002 (2019).
- [6] P. Mehta, M. Bukov, C.-H. Wang, A. G. Day, C. Richardson, C. K. Fisher, and D. J. Schwab, A high-bias, low-variance introduction to machine learning for physicists, *Phys. Rep.* **810**, 1 (2019).
- [7] Z. Li, J. Yoon, R. Zhang, F. Rajabipour, W. V. Srubar III, I. Dabo, and A. Radlińska, Machine learning in concrete science: Applications, challenges, and best practices, *npj Comput. Mater.* **8**, 1 (2022).
- [8] S. L. Wu, J. Chan, W. Guan, S. Sun, A. Wang, C. Zhou, M. Livny, F. Carminati, A. Di Meglio, A. C. Li *et al.*, Application of quantum machine learning using the quantum variational classifier method to high energy physics analysis at the LHC on IBM quantum computer simulator and hardware with 10 qubits, *J. Phys. G: Nucl. Part. Phys.* **48**, 125003 (2021).
- [9] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces, *Nature (London)* **567**, 209 (2019).
- [10] M. Schuld and N. Killoran, Quantum Machine Learning in Feature Hilbert Spaces, *Phys. Rev. Lett.* **122**, 040504 (2019).
- [11] Y. Liu, S. Arunachalam, and K. Temme, A rigorous and robust quantum speed-up in supervised machine learning, *Nat. Phys.* **17**, 1013 (2021).
- [12] J. A. Keith, V. Vassilev-Galindo, B. Cheng, S. Chmiela, M. Gastegger, K.-R. Müller, and A. Tkatchenko, Combining machine learning and computational chemistry for predictive insights into chemical systems, *Chem. Rev.* **121**, 9816 (2021).
- [13] M. Sajjan, J. Li, R. Selvarajan, S. H. Sureshababu, S. S. Kale, R. Gupta, V. Singh, and S. Kais, Quantum machine learning for chemistry and physics, *Chem. Soc. Rev.* **51**, 6475 (2022).
- [14] B. Swingle, Unscrambling the physics of out-of-time-order correlators, *Nat. Phys.* **14**, 988 (2018).
- [15] B. Hall, Matrix Lie groups, in *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction* (Springer International Publishing, Cham, 2015), pp. 3–30.
- [16] B. Swingle, G. Bentsen, M. Schleier-Smith, and P. Hayden, Measuring the scrambling of quantum information, *Phys. Rev. A* **94**, 040302(R) (2016).
- [17] T. Hashizume, G. S. Bentsen, S. Weber, and A. J. Daley, Deterministic Fast Scrambling with Neutral Atom Arrays, *Phys. Rev. Lett.* **126**, 200603 (2021).
- [18] C.-J. Lin and O. I. Motrunich, Out-of-time-ordered correlators in a quantum Ising chain, *Phys. Rev. B* **97**, 144304 (2018).
- [19] S. Pegahan, I. Arakelyan, and J. E. Thomas, Energy-Resolved Information Scrambling in Energy-Space Lattices, *Phys. Rev. Lett.* **126**, 070601 (2021).
- [20] M. Brenes, S. Pappalardi, M. T. Mitchison, J. Goold, and A. Silva, Out-of-time-order correlations and the fine structure of eigenstate thermalization, *Phys. Rev. E* **104**, 034120 (2021).
- [21] R. Fan, P. Zhang, H. Shen, and H. Zhai, Out-of-time-order correlation for many-body localization, *Sci. Bull.* **62**, 707 (2017).
- [22] Z.-H. Sun, J. Cui, and H. Fan, Quantum information scrambling in the presence of weak and strong thermalization, *Phys. Rev. A* **104**, 022405 (2021).
- [23] K. K. Sharma and V. P. Gerdt, Quantum information scrambling and entanglement in bipartite quantum states, *Quant. Info. Proc.* **20**, 1 (2021).
- [24] A. Touil and S. Deffner, Quantum scrambling and the growth of mutual information, *Quantum Sci. Technol.* **5**, 035005 (2020).
- [25] B. Yan, L. Cincio, and W. H. Zurek, Information Scrambling and Loschmidt Echo, *Phys. Rev. Lett.* **124**, 160603 (2020).
- [26] W. Fu and S. Sachdev, Numerical study of fermion and boson models with infinite-range random interactions, *Phys. Rev. B* **94**, 035135 (2016).
- [27] C. Yin and A. Lucas, Bound on quantum scrambling with all-to-all interactions, *Phys. Rev. A* **102**, 022402 (2020).

- [28] E. Iyoda and T. Sagawa, Scrambling of quantum information in quantum many-body systems, *Phys. Rev. A* **97**, 042330 (2018).
- [29] S. Sachdev, Bekenstein-Hawking Entropy and Strange Metals, *Phys. Rev. X* **5**, 041025 (2015).
- [30] X. Mi, P. Roushan, C. Quintana, S. Mandra, J. Marshall, C. Neill, F. Arute, K. Arya, J. Atalaya, R. Babbush *et al.*, Information scrambling in quantum circuits, *Science* **374**, 1479 (2021).
- [31] Q. Zhu, Z.-H. Sun, M. Gong, F. Chen, Y.-R. Zhang, Y. Wu, Y. Ye, C. Zha, S. Li, S. Guo, H. Qian, H.-L. Huang, J. Yu, H. Deng, H. Rong, J. Lin, Y. Xu, L. Sun, C. Guo, N. Li *et al.*, Observation of Thermalization and Information Scrambling in a Superconducting Quantum Processor, *Phys. Rev. Lett.* **128**, 160502 (2022).
- [32] M. S. Blok, V. V. Ramasesh, T. Schuster, K. O'Brien, J. M. Kreikebaum, D. Dahlen, A. Morvan, B. Yoshida, N. Y. Yao, and I. Siddiqi, Quantum Information Scrambling on a Superconducting Qutrit Processor, *Phys. Rev. X* **11**, 021010 (2021).
- [33] K. A. Landsman, C. Figgatt, T. Schuster, N. M. Linke, B. Yoshida, N. Y. Yao, and C. Monroe, Verified quantum information scrambling, *Nature (London)* **567**, 61 (2019).
- [34] J. Harris, B. Yan, and N. A. Sinitsyn, Benchmarking Information Scrambling, *Phys. Rev. Lett.* **129**, 050602 (2022).
- [35] R. G. Melko, G. Carleo, J. Carrasquilla, and J. I. Cirac, Restricted Boltzmann machines in quantum physics, *Nat. Phys.* **15**, 887 (2019).
- [36] G. E. Hinton and R. R. Salakhutdinov, Reducing the dimensionality of data with neural networks, *Science* **313**, 504 (2006).
- [37] G. Torlai and R. G. Melko, Learning thermodynamics with Boltzmann machines, *Phys. Rev. B* **94**, 165134 (2016).
- [38] A. Argun, A. Callegari, and G. Volpe, Ising model, in *Simulation of Complex Systems* (IOP Publishing, Bristol, 2021), pp. 2–1 to 2–12.
- [39] S. G. Brush, History of the Lenz-Ising model, *Rev. Mod. Phys.* **39**, 883 (1967).
- [40] G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo, Many-body quantum state tomography with neural networks, *Nat. Phys.* **14**, 447 (2018).
- [41] K. Ch'Ng, J. Carrasquilla, R. G. Melko, and E. Khatami, Machine Learning Phases of Strongly Correlated Fermions, *Phys. Rev. X* **7**, 031038 (2017).
- [42] A. Decelle and C. Furtlehner, Exact Training of Restricted Boltzmann Machines on Intrinsically Low Dimensional Data, *Phys. Rev. Lett.* **127**, 158303 (2021).
- [43] J. Carrasquilla and R. G. Melko, Machine learning phases of matter, *Nat. Phys.* **13**, 431 (2017).
- [44] M. Sajjan, S. H. Sureshbabu, and S. Kais, Quantum machine-learning for eigenstate filtration in two-dimensional materials, *J. Am. Chem. Soc.* **143**, 18426 (2021).
- [45] R. Xia and S. Kais, Quantum machine learning for electronic structure calculations, *Nat. Commun.* **9**, 4195 (2018).
- [46] S. Kanno and T. Tada, Many-body calculations for periodic materials via restricted Boltzmann machine-based VQE, *Quantum Sci. Technol.* **6**, 025015 (2021).
- [47] M. Razavy, *Heisenberg's Quantum Mechanics* (World Scientific, Singapore, 2011).
- [48] K. Hashimoto, K. Murata, and R. Yoshii, Out-of-time-order correlators in quantum mechanics, *J. High Energy Phys.* **10** (2017) 138.
- [49] H. Shen, P. Zhang, R. Fan, and H. Zhai, Out-of-time-order correlation at a quantum phase transition, *Phys. Rev. B* **96**, 054503 (2017).
- [50] J. Braumüller, A. H. Karamlou, Y. Yanay, B. Kannan, D. K. Kim, M. Kjaergaard, A. Melville, B. M. Niedzielski, Y. Sung, A. Vepsäläinen *et al.*, Probing quantum information propagation with out-of-time-ordered correlators, *Nat. Phys.* **18**, 172 (2022).
- [51] B. Sundar, A. Elben, L. K. Joshi, and T. V. Zache, Proposal for measuring out-of-time-ordered correlators at finite temperature with coupled spin chains, *New J. Phys.* **24**, 023037 (2022).
- [52] L. D'Alessio, Y. Kafri, A. Polkovnikov, and M. Rigol, From quantum chaos and eigenstate thermalization to statistical mechanics and thermodynamics, *Adv. Phys.* **65**, 239 (2016).
- [53] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, Cambridge, 2010).
- [54] J. Watrous, *The Theory of Quantum Information* (Cambridge University Press, Cambridge, 2018).
- [55] A. Globerson, E. Stark, E. Vaadia, and N. Tishby, The minimum information principle and its application to neural code analysis, *Proc. Natl. Acad. Sci. USA* **106**, 3490 (2009).
- [56] G. De Tomasi, S. Bera, J. H. Bardarson, and F. Pollmann, Quantum Mutual Information as a Probe for Many-Body Localization, *Phys. Rev. Lett.* **118**, 016804 (2017).
- [57] M. M. Wolf, F. Verstraete, M. B. Hastings, and J. I. Cirac, Area Laws in Quantum Systems: Mutual Information and Correlations, *Phys. Rev. Lett.* **100**, 070502 (2008).
- [58] G. Carleo and M. Troyer, Solving the quantum many-body problem with artificial neural networks, *Science* **355**, 602 (2017).
- [59] S. Sorella, M. Casula, and D. Rocca, Weak binding between two aromatic rings: Feeling the van der Waals attraction by quantum Monte Carlo methods, *J. Chem. Phys.* **127**, 014105 (2007).
- [60] F. Vicentini, D. Hofmann, A. Szabó, D. Wu, C. Roth, C. Giuliani, G. Pescia, J. Nys, V. Vargas-Caldern, N. Astrakhantsev, and G. Carleo, NetKet 3: Machine learning toolbox for many-body quantum systems, *SciPost Phys. Codebases*, **7** (2022).
- [61] G. Vitagliano, A. Riera, and J. I. Latorre, Volume-law scaling for the entanglement entropy in spin-1/2 chains, *New J. Phys.* **12**, 113049 (2010).
- [62] T. Koffel, M. Lewenstein, and L. Tagliacozzo, Entanglement Entropy for the Long-Range Ising Chain in a Transverse Field, *Phys. Rev. Lett.* **109**, 267203 (2012).
- [63] T. Kuwahara and K. Saito, Area law of noncritical ground states in 1d long-range interacting systems, *Nat. Commun.* **11**, 4478 (2020).
- [64] D. Sehayek, A. Golubeva, M. S. Albergo, B. Kulchytskyy, G. Torlai, and R. G. Melko, Learnability scaling of quantum states: Restricted Boltzmann machines, *Phys. Rev. B* **100**, 195125 (2019).
- [65] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 2nd ed. (Cambridge University Press, Cambridge, 2012).
- [66] P. Pfeuty, The one-dimensional ising model with a transverse field, *Ann. Phys.* **57**, 79 (1970).
- [67] P. Pfeuty and R. Elliott, The Ising model with a transverse field. II. Ground state properties, *J. Phys. C* **4**, 2370 (1971).
- [68] R. Stinchcombe, Ising model in a transverse field. i. basic theory, *J. Phys. C* **6**, 2459 (1973).

- [69] Y. Nomura, A. S. Darmawan, Y. Yamaji, and M. Imada, Restricted Boltzmann machine learning for solving strongly correlated quantum systems, *Phys. Rev. B* **96**, 205152 (2017).
- [70] K. Choo, A. Mezzacapo, and G. Carleo, Fermionic neural-network states for *ab-initio* electronic structure, *Nat. Commun.* **11**, 2368 (2020).
- [71] S. Lu, X. Gao, and L.-M. Duan, Efficient representation of topologically ordered states with restricted Boltzmann machines, *Phys. Rev. B* **99**, 155136 (2019).
- [72] M. Sajjan, H. Alaeian, and S. Kais, Magnetic phases of spatially modulated spin-1 chains in Rydberg excitons: Classical and quantum simulations, *J. Chem. Phys.* **157**, 224111 (2022).
- [73] T. Viejra, C. Casert, J. Nys, W. De Neve, J. Haegeman, J. Ryckebusch, and F. Verstraete, Restricted Boltzmann Machines for Quantum States with Non-Abelian or Anyonic Symmetries, *Phys. Rev. Lett.* **124**, 097201 (2020).
- [74] C. K. Lee, P. Patil, S. Zhang, and C. Y. Hsieh, Neural-network variational quantum algorithm for simulating many-body dynamics, *Phys. Rev. Res.* **3**, 023095 (2021).
- [75] S. H. Sureshbabu, M. Sajjan, S. Oh, and S. Kais, Implementation of quantum machine learning for electronic structure calculations of periodic systems on quantum computing devices, *J. Chem. Inf. Model.* **61**, 2667 (2021).
- [76] P.-J. Yang, M. Sugiyama, K. Tsuda, and T. Yanai, Artificial neural networks applied as molecular wave function solvers, *J. Chem. Theory Comput.* **16**, 3513 (2020).
- [77] C. Ciliberto, M. Herbster, A. D. Ialongo, M. Pontil, A. Rocchetto, S. Severini, and L. Wossnig, Quantum machine learning: A classical perspective, *Proc. R. Soc. A: Math. Phys. Engn. Sci.* **474**, 20170551 (2018).
- [78] N. Le Roux and Y. Bengio, Representational power of restricted Boltzmann machines and deep belief networks, *Neural Comput.* **20**, 1631 (2008).
- [79] D.-L. Deng, X. Li, and S. Das Sarma, Quantum Entanglement in Neural Network States, *Phys. Rev. X* **7**, 021021 (2017).
- [80] P. M. Long and R. A. Servedio, Restricted Boltzmann machines are hard to approximately evaluate or simulate, in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10 (Omnipress, Madison, WI, 2010), pp. 703–710.
- [81] H. Shen, P. Zhang, Y.-Z. You, and H. Zhai, Information Scrambling in Quantum Neural Networks, *Phys. Rev. Lett.* **124**, 200504 (2020).
- [82] Y. Wu, P. Zhang, and H. Zhai, Scrambling ability of quantum neural network architectures, *Phys. Rev. Res.* **3**, L032057 (2021).
- [83] R. J. Garcia, K. Bu, and A. Jaffe, Quantifying scrambling in quantum neural networks, *J. High Energy Phys.* **03** (2022) 027.
- [84] T. Schuster, M. Niu, J. Cotler, T. O'Brien, J. R. McClean, and M. Mohseni, Learning quantum systems via out-of-time-order correlators, [arXiv:2208.02254](https://arxiv.org/abs/2208.02254).
- [85] M. Gärttner, J. G. Bohnet, A. Safavi-Naini, M. L. Wall, J. J. Bollinger, and A. M. Rey, Measuring out-of-time-order correlations and multiple quantum spectra in a trapped-ion quantum magnet, *Nat. Phys.* **13**, 781 (2017).
- [86] Z. Holmes, A. Arrasmith, B. Yan, P. J. Coles, A. Albrecht, and A. T. Sornborger, Barren Plateaus Preclude Learning Scramblers, *Phys. Rev. Lett.* **126**, 190501 (2021).
- [87] C. Bellinger, R. Coles, M. Crowley, and I. Tambllyn, Reinforcement learning in a physics-inspired Semi-Markov environment, in *Advances in Artificial Intelligence. Canadian AI 2020, Lecture Notes in Computer Science*, Vol. 12109 (Springer, Cham, 2020), pp. 55–66.
- [88] F. Musil, A. Grisafi, A. P. Bartók, C. Ortner, G. Csányi, and M. Ceriotti, Physics-inspired structural representations for molecules and materials, *Chem. Rev.* **121**, 9759 (2021).
- [89] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, Physics-informed machine learning, *Nat. Rev. Phys.* **3**, 422 (2021).
- [90] A. Fachechi, E. Agliari, and A. Barra, Dreaming neural networks: Forgetting spurious memories and reinforcing pure ones, *Neural Networks* **112**, 24 (2019).
- [91] E. Agliari, F. Alemanno, A. Barra, and A. Fachechi, Dreaming neural networks: Rigorous results, *J. Stat. Mech.* (2019) 083503.
- [92] M. Aquaro, F. Alemanno, I. Kanter, F. Durante, E. Agliari, and A. Barra, Recurrent neural networks that generalize from examples and optimize by dreaming, [arXiv:2204.07954](https://arxiv.org/abs/2204.07954).
- [93] D. Bank, N. Koenigstein, and R. Giryes, Autoencoders, [arXiv:2003.05991](https://arxiv.org/abs/2003.05991) (2020).