Convolutional neural networks for large-scale dynamical modeling of itinerant magnets

Xinlun Cheng ,[1,2] Sheng Zhang ,[3] Phong C. H. Nguyen ,[1] Shahab Azarfar ,[1] Gia-Wei Chern,[3] and Stephen S. Baek [1,4]

[1] *School of Data Science, University of Virginia, Charlottesville, Virginia 22903, USA*
[2] *Department of Astronomy, University of Virginia, Charlottesville, Virginia 22904, USA*
[3] *Department of Physics, University of Virginia, Charlottesville, Virginia 22904, USA*
[4] *Department of Mechanical and Aerospace Engineering, University of Virginia, Charlottesville, Virginia 22903, USA*

(Received 30 June 2023; accepted 24 August 2023; published 14 September 2023)

Complex spin textures in itinerant electron magnets hold promises for next-generation memory and information technology. The long-ranged and often frustrated electron-mediated spin interactions in these materials give rise to intriguing localized spin structures such as skyrmions. Yet, simulations of magnetization dynamics for such itinerant magnets are computationally difficult due to the need for repeated solutions to the electronic structure problems. We present a convolutional neural network (CNN) model to accurately and efficiently predict the electron-induced magnetic torques acting on local spins. Importantly, as the convolutional operations with a fixed kernel (receptive field) size naturally take advantage of the locality principle for many-electron systems, CNNs offer a scalable machine learning approach to spin dynamics. We apply our approach to enable large-scale dynamical simulations of skyrmion phases in itinerant spin systems. By incorporating the CNN model into Landau-Lifshitz-Gilbert dynamics, our simulations successfully reproduce the relaxation process of the skyrmion phase and stabilize a skyrmion lattice in larger systems. The CNN model also allows us to compute the effective receptive fields, thus providing a systematic and unbiased method for determining the locality of the original electron models.

DOI: 10.1103/PhysRevResearch.5.033188

I. INTRODUCTION

Itinerant frustrated magnets with electron-mediated spin-spin interactions often exhibit complex noncollinear or noncoplanar spin textures. Of particular interest are particle-like objects such as magnetic vortices and skyrmions which are not only of fundamental interest in magnetism but also have important technological implications in the emerging field of *spintronics* [1–7]. These nanometer-sized localized spin textures are characterized by nontrivial topological invariants and are rather stable objects with long lifetimes. In itinerant electron magnets, skyrmions can be moved, created, and annihilated not only by magnetic fields but also by electrical currents due to electron-spin interactions. The presence of such complex textures could also give rise to intriguing electronic and transport properties, such as the topological Hall effects and topological Nernst effects [7–10], due to a nontrivial Berry phase acquired by electrons when traversing over closed loops of noncoplanar spins [11].

Dynamical modeling of complex textures in itinerant spin systems, however, is a computationally challenging task. While magnetic moments in most metallic skyrmion materials can be well approximated as classical spin vectors, the local effective magnetic fields, analogous to forces in molecular dynamics, originate from exchange interactions with itinerant electrons and must be computed quantum mechanically. Dynamics simulations of such itinerant magnets thus require solving an electronic structure problem associated with the instantaneous spin configuration at every time step. Repeated quantum calculations would be prohibitively expensive for large-scale simulations. Consequently, empirical classical spin Hamiltonians, from which the local fields can be explicitly calculated, are often employed in large-scale dynamical simulations of skyrmion magnets [12,13]. Yet, such classical spin models often cannot capture the intricate long-range spin-spin interactions mediated by electrons.

The computational complexity of the above quantum approaches to spin dynamics is similar to the so-called quantum or *ab initio* molecular dynamics (MD) methods. Contrary to classical MD methods that are based on empirical force fields, the atomic forces in quantum MD are computed by integrating out electrons on the fly as the atomic trajectories are generated [14]. Various many-body methods, notably the density functional theory, have been used for the force calculation of quantum MD. However, the computational cost of repeated electronic structure solutions significantly restricts the accessible scales of atomic simulations. To overcome this computational difficulty, machine learning (ML) methods have been exploited to develop force-field models by accurately emulating the time-consuming many-body calculations, thus enabling large-scale MD simulations with the desired quantum accuracy.

Crucial to the remarkable scalability of ML-based force-field models is the divide-and-conquer approach proposed in the pioneering works of Behler and Parrinello [15] and Bartók *et al.* [16]. In this approach, the total energy of the system is partitioned into local contributions $E = \sum_i \epsilon_i$, where ϵ_i

is called the atomic energy and depends on only the local environment of the ith atom [15,16]. The atomic forces are then obtained from the derivatives of the predicted energy, $\mathbf{F}_i = -\partial E/\partial \mathbf{r}_i$, where $\mathbf{r}_i$ is the atomic position vector. Crucially, the complicated dependence of atomic energy ϵ_i on its local neighborhood is approximated by the ML model, which is trained on the condition that both the predicted individual forces $\mathbf{F}_i$ as well as the total energy E agree with the quantum calculations [15–26]. It is worth noting that physically the principle of locality, or the so-called nearsightedness of electronic matters, lies at the heart of this approach [27,28].

The tremendous success of ML methods in quantum MD simulations has spurred similar approaches to multiscale dynamical modeling of other functional electronic systems in condensed matter physics [29–35]. In particular, the Behler-Parrinello (BP) ML scheme [15,16] was generalized to build effective magnetic energy or torque-field models with the accuracy of quantum calculations for itinerant electron magnets [33,34,36,37]. Notably, large-scale dynamical simulations enabled by such ML models uncovered intriguing phase separation dynamics that results from the nontrivial interplay between electrons and local spins. While the conventional BP scheme can only represent conservative forces, a generalized potential theory for the Landau-Lifshitz equation allows one to extend the BP scheme to describe nonconserved spin torques that are crucial to the dynamical modeling of out-of-equilibrium itinerant spin systems [35].

In this paper we present an ML torque model for itinerant magnets based on convolutional neural networks (CNNs). CNNs are a class of neural networks that can be characterized by its local connectivity, implemented via finite-sized convolution kernels. Importantly, the convolution operation with a finite-sized kernel naturally incorporates the locality principle into the ML structure, thus offering an efficient implementation of the ML torque model that can be straightforwardly scaled to larger systems. Our CNN model is designed to directly predict the vector torque field at every site without the need for the introduction of local energies as in the BP scheme. Data augmentation techniques are employed to incorporate the spin-rotational symmetry and the lattice symmetry into the CNN spin-torque model. We demonstrate our approach on an itinerant spin model which exhibits a skyrmion crystal phase at an intermediate magnetic field. We show that dynamical simulations with magnetic torques computed from the trained CNN model faithfully reproduce the relaxation process of the itinerant spin systems. Moreover, the CNN model, while trained by datasets from small systems, is capable of stabilizing a skyrmion lattice on larger systems, thus demonstrating the transferability and scalability of our ML approach.

The rest of the paper is organized as follows. In Sec. II we discuss the methods for simulating the spin dynamics of itinerant electron magnets. A triangular-lattice *s*-*d* model, a well-studied itinerant spin system, is used as a concrete example to highlight the complexity of the dynamical simulations. We also briefly review BP-type ML approaches, where a neural network is trained to approximate a local energy function. Section III presents the CNN structure used for the prediction of spin torque. Details of the data augmentation for incorporating symmetries and how the ML model can be scaled to larger systems are also discussed. Using the *s*-*d* model as an example, a benchmark of the CNN models and simulation results based on the trained models are presented in Sec. IV. We also ascertain the scalability and symmetry of the proposed CNN method, as well as its compliance with the locality principle. Finally, we summarize our work and discuss future directions in Sec. V.

II. MAGNETIZATION DYNAMICS OF THE ITINERANT MAGNETS

The magnetization dynamics in spin systems is governed by the Landau-Lifshitz-Gilbert (LLG) equation [38]

$$\frac{d\mathbf{S}_i}{dt} = \mathbf{T}_i - \alpha\, \mathbf{S}_i \times \mathbf{T}_i + \boldsymbol{\tau}_i, \tag{1}$$

where $\mathbf{T}_i$ is the magnetic torque defined as

$$\mathbf{T}_i = \gamma \mathbf{S}_i \times \mathbf{H}_i. \tag{2}$$

Here γ is the gyromagnetic ratio, and $\mathbf{H}_i$ is an effective exchange field acting on spin-i, α is the damping coefficient, and $\boldsymbol{\tau}_i(t) = \mathbf{S}_i \times \boldsymbol{\eta}_i(t)$ is a fluctuating torque generated by a random local field $\boldsymbol{\eta}_i$ of zero mean. The stochastic field $\boldsymbol{\eta}_i$ is assumed to be a Gaussian random variable with the variance determined from α and temperature T through the fluctuation-dissipation theorem. The LLG simulations are widely used to study dynamical phenomena in a wide range of magnetic systems, including spin waves in unusual magnetic phases and dynamical behaviors of skyrmions and other spin textures.

For adiabatic spin dynamics, the local exchange field is given by the derivative of the system energy $E = E(\mathbf{S}_i)$:

$$\mathbf{H}_i = -\frac{\partial E}{\partial \mathbf{S}_i}. \tag{3}$$

For magnetic insulators, interactions between spins are often short-ranged. The resultant magnetic energy has the form of bilinear interactions between a few nearest-neighbor spins on the lattice, e.g., $E = \sum_{ij}(J_{ij}\mathbf{S}_i \cdot \mathbf{S}_j + \mathbf{D}_{ij} \cdot \mathbf{S}_i \times \mathbf{S}_j)$, where J_{ij} denotes the isotropic Heisenberg exchange interaction and $\mathbf{D}_{ij}$ represents the anisotropic exchange, also known as the Dzyaloshinskii-Moriya interaction [12,13]. The exchange field of such models is explicitly given by $\mathbf{H}_i = -\sum_j(J_{ij}\mathbf{S}_j + \mathbf{D}_{ij} \times \mathbf{S}_j)$, where the summation is restricted to a few nearest neighbors, and can be very efficiently computed for large-scale LLG simulations.

On the other hand, the exchange fields in a metallic magnet come from interactions between local spins and itinerant electrons. Here we consider spin dynamics in the adiabatic approximation, which is analogous to the Born-Oppenheimer approximation in quantum molecular dynamics [14]. In the adiabatic limit, electron relaxation is assumed to be much faster than the timescale of local magnetic moments. As a result, the magnetic energy E in Eq. (3) can be obtained by freezing the spin configuration and integrating out the electrons. The resultant spin-dependent energy function, $E = E(\mathbf{S}_i)$, can be viewed as a potential energy surface (PES) in the high-dimensional spin space, similar to the PES in Born-Oppenheimer MD simulations. In practice, the calculation of this magnetic PES requires solving the electron structure problem that depends on the instantaneous spin structure $\{\mathbf{S}_i(t)\}$.

For concreteness, here we consider a generic single-band *s-d* model for such itinerant magnets. The *s-d* model describes the interaction between itinerant *s*-band electrons and magnetic moments $\mathbf{S}_i$ of localized *d*-electrons. Its Hamiltonian reads

$$\mathcal{H} = \sum_{ij} \sum_{\alpha=\uparrow,\downarrow} t_{ij} c^{\dagger}_{i\alpha} c_{j\alpha} - J \sum_{i} \sum_{\alpha,\beta=\uparrow,\downarrow} \mathbf{S}_i \cdot c^{\dagger}_{i\alpha} \boldsymbol{\sigma}_{\alpha\beta} c_{i\beta}, \quad (4)$$

where $c^{\dagger}_{i\alpha}/c_{i,\alpha}$ are creation/annihilation operators of an electron with spin $\alpha =\uparrow, \downarrow$ at site i, t_{ij} is the electron hopping constant between a pair of sites (i, j), J denotes the strength of local Hund's rule coupling between electron spin and magnetic moment $\mathbf{S}_i$ of localized d-electrons. For most skyrmion magnets, these local magnetic moments can be well approximated as classical spins of fixed length $|\mathbf{S}_i| = S$.

For small Hund's coupling $J \ll t_{ij}$, the effective energy of spins can be obtained by integrating out electrons via a second-order perturbation calculation, giving rise to a long-ranged oscillatory interaction, similar to the so-called Ruderman-Kittel-Kasuya-Yosida (RKKY) interaction [39–41]. However, for intermediate and large Hund's coupling, the effective energy to be used for the force calculation in Eq. (3) has to be obtained by integrating out electrons on the fly:

$$E = \langle \mathcal{H} \rangle = \mathrm{Tr}(\rho \mathcal{H}), \quad (5)$$

where $\rho = \exp(-\mathcal{H}/k_B T)$ is the density matrix of the equilibrium electron liquid within the adiabatic approximation. The calculation of the density matrix, in the absence of electron-electron interaction, amounts to solving a disordered tight-binding Hamiltonian for a given spin configuration. The standard method for solving tight-binding models is based on exact diagonalization, whose complexity scales cubically with the system size. As a result, for large-scale LLG simulations of the *s-d* model, repeated ED calculations of the electron density matrix can be overwhelmingly time-consuming.

As discussed in Sec. I, the BP scheme has been generalized to develop ML-based models for the effective spin energy $E(\{\mathbf{S}_i\})$ of itinerant magnets [33–37]. In this approach, the total energy is partitioned into local contributions

$$E = \sum_i \epsilon_i = \sum_i \varepsilon(\mathcal{C}_i), \quad (6)$$

where the energy $\epsilon_i = \varepsilon(\mathcal{C}_i)$ is associated with the ith lattice site and is assumed to depend only on spin configuration $\mathcal{C}_i = \{\mathbf{S}_j \mid \|\mathbf{r}_j - \mathbf{r}_i\| < r_c\}$ in its neighborhood. This local energy function $\varepsilon(\mathcal{C}_i)$ can be viewed as the building block of the magnetic PES. Importantly, the complicated dependence of the PES on the neighborhood spins is to be approximated by fully connected neural networks [33–35]. To preserve the SO(3) spin rotation symmetry, the inner product between spin pairs $b_{jk} = \mathbf{S}_j \cdot \mathbf{S}_k$ and scalar product between spin triplets $\chi_{jkl} = \mathbf{S}_j \cdot \mathbf{S}_k \times \mathbf{S}_l$ within the neighborhood are used as building blocks to construct feature variables that are input to the neural network. Finally, exchange fields $\mathbf{H}_i$ acting on spins are obtained by applying automatic differentiation to the ML energy model.

III. CNN SPIN TORQUE MODEL

The BP-type schemes described here essentially provide an energy-based ML model for force field calculations. A crucial step is the partitioning of the total energy into local contributions ϵ_i, which cannot be directly computed from electronic structure methods that are used to generate the training dataset. As a result, the loss function L cannot be directly determined from the predicted energies ϵ_i. Instead, it is constructed from the, e.g., mean square error (MSE) or "forces," or in our case, the spin torque fields, and only implicitly depends on the predicted energy through automatic differentiation. However, the uncertainties due to the introduction of such intermediate local energies often complicate the training of BP-type models. While one advantage of the BP-type scheme is the explicit inclusion of the physical constraint of conservative forces, such energy-based ML approaches, however, are also restricted to the representation of only conservative forces. In this section we present an alternative ML approach that directly predicts the vector forces without going through intermediate energy.

A. Convolutional neural networks

The fact that spins in metallic magnets are defined on well-known lattices suggests that spin configurations can be treated as generalized "images," which can then be processed using powerful image-processing techniques developed in recent years, such as CNNs. Below, we present a CNN model for the direct prediction of torques $\mathbf{T}_i$ that drive the spin dynamics. As illustrated in Fig. 1, the proposed network takes the spin configuration $\{\mathbf{S}_i\}$ on the lattice as input and returns the torques $\{\mathbf{T}_i\}$ that drive the spin dynamics as output. The model comprises multiple convolution layers f_m with associated activation (nonlinearity) layers σ_m to model the complex nonlinear relationship between $\{\mathbf{S}_i\}$ and $\{\mathbf{T}_i\}$ as a composition of such layers, $f_{\mathrm{CNN}} = (\sigma_L \circ f_L) \circ \cdots \circ (\sigma_1 \circ f_1)$, where L is the number of layers or the *depth* of the CNN model.

Given an input vector field $V \in C^{\infty}(\mathbb{R}^2, \mathbb{R}^d)$, each convolution layer f_m maps the vector field onto an output vector field $W \in C^{\infty}(\mathbb{R}^2, \mathbb{R}^k)$ by convolving a *kernel* tensor field $h_m(X) := h(X; \theta_m)$ with trainable parameters θ_m, via the convolution operation:

$$W(\mathbf{r}) := \int_{\mathbb{R}^2} V(\mathbf{q}) h_m(\mathbf{r} - \mathbf{q})\, d\mathbf{q}. \quad (7)$$

Each vector element of the vector field W then undergoes the activation function $\sigma_m : \mathbb{R} \to \mathbb{R}$ to produce the output vector field $A \in C^{\infty}(\mathbb{R}^2, \mathbb{R}^k)$, called *activation maps*. A variety of activation functions can be used in CNNs. In the current work, we use the *rectified linear unit*, or *ReLU* [42] as an activation function,

$$\sigma_m(x) := \max(0, x), \quad (8)$$

for $m = 1, \ldots, L - 1$. Note that the final layer f_L has no activation function associated with it, or technically, $\sigma_L(x) = x$.

Typically in CNNs, the support of a kernel $\mathrm{supp}(h_m)$, i.e., the region where h_m has nonzero values (also known as the *receptive field* of the kernel), is limited to a small region (e.g., 5 × 5 lattice sites) such that the activation response $W(\mathbf{r})$, thereby $A(\mathbf{r})$, at position $\mathbf{r}$ is limited to the patterns of V only

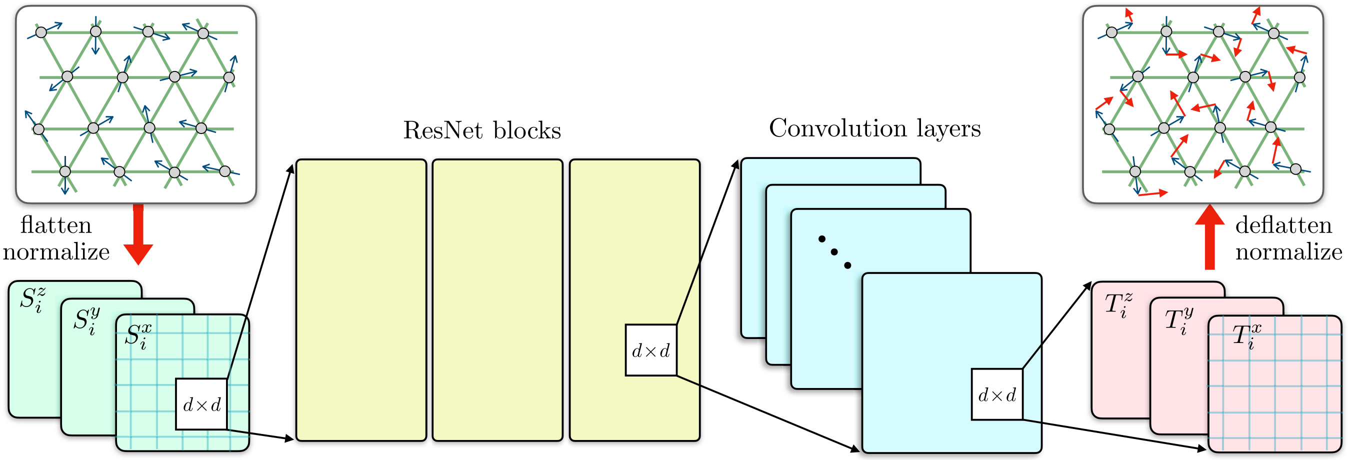

FIG. 1. Schematic diagram of the CNN-based ML model for spin-torque prediction of itinerant electron magnets. The spin configuration $\{\mathbf{S}_i\}$ on a lattice is first flattened to give three arrays, corresponding to the three components of spins, which are input to a series of ResNet blocks. Details of the ResNet are presented in Fig. 2. The output of the ResNet blocks is then processed through additional convolution layers. The final output are three arrays which after deflattening correspond to the torques $\{\mathbf{T}_i\}$ that drive the spin dynamics.

within the close proximity of $\mathbf{r}$. It is worth noting that the physical justification of employing such finite-size kernels is the principle of locality: *viz.,* local physical quantities, such as local spin torque $\mathbf{T}_i = \mathbf{T}(\mathbf{r}_i)$, are predominately determined by the local environment of site i:

$$\mathbf{T}_i = \boldsymbol{\mathcal{T}}(\mathcal{C}_i), \tag{9}$$

where $\mathcal{C}_i$ is the magnetic environment in the vicinity of site i, and the vector function $\boldsymbol{\mathcal{T}}(\cdot)$ is to be modeled by the CNN. The range of the neighborhood $\mathcal{C}_i$ is determined by the sizes of kernels and the number of convolution layers.

Note that Eqs. (7) and (8) imply that the output activation $A(\mathbf{r})$ at position $\mathbf{r}$ will be of a large, positive magnitude, only if the input vector field $V(\mathbf{r})$ is closely correlated to the (shifted) kernel $h_m(\mathbf{q} - \mathbf{r})$. Therefore, the goal of training the CNN is to find the unknown kernel parameters $\theta_{m=1,\ldots,L}$ that determines the function shapes of $\{h_m\}$, in order to produce the adequate activation values such that the final output of the model $f_{\mathrm{CNN}}(\{\mathbf{S}_i\})$ can reasonably approximate the ground truth spin torque $\{\mathbf{T}_i\}$ in the training data.

Meanwhile, the composition of convolution layers enables hierarchical modeling of the spin-torque relationship. That is, while an individual kernel limited to a small region may represent only rather simplistic patterns (e.g., small blobs), the composition of such kernels across layers alongside the nonlinear ReLU activation can produce fairly complex, nonlinear patterns. Furthermore, the composition of convolution layers $(\sigma_n \circ f_n) \circ (\sigma_m \circ f_m)$ in effect produces a larger receptive field area, equal to the Minkowski sum[1] of the receptive fields of the individual layers $\mathrm{supp}(h_n \otimes h_m) = \mathrm{supp}(h_n) \oplus \mathrm{supp}(h_m)$. Therefore, stacked convolution layers produce a natural hierarchy, in which earlier layers represent local, primitive patterns in a small proximity while latter (deeper) layers represent more global, sophisticated patterns in a relatively larger periphery.

Finally, a purely convolutional CNN, without any conventional fully connected (dense) layers, can restrict the overall receptive field size of the entire model $\mathrm{supp}(h_L \otimes \cdots \otimes h_1)$ to a predetermined lattice size, presenting a distinct advantage of *built-in locality*. Further, with a purely convolutional design, since the sizes of kernels in a CNN are fixed for a given itinerant model, the successfully trained CNN model can then be used in much larger lattice systems without the need to rebuild or retrain a new neural network. The CNN structure thus provides a natural approach to implementing scalable ML models based on the locality principle.

B. Model architecture

Figure 1 shows a schematic diagram of our CNN architecture. The input to the CNN is the spin vector field $\{\mathbf{S}_i\}$ transformed from triangular lattice to square ("flattening") as most convolution operation expects a square input. We employed four ResNet blocks, inspired by He *et al.* [43], as our backbone, which processes the input into an activation map of 512 features. These features then undergo two additional convolution layers, which in the end produce the torques as the output of the network. In our model these torque vector outputs are obtained in a normalized range of values with a mean magnitude of 1. Such normalized outputs are then scaled by the mean magnitude of the torque vectors in the training data set. Finally, the predicted torques on the square grid are "unflattened" onto the original triangular lattice.

Compared to fully connected (dense) layers, in which each neuron aggregates values across the entire domain into a single scalar value via the weighted sum, convolution layers preserve the spatial structure of the input domain. Therefore, with the proper boundary condition (e.g., "padding" in the machine learning jargon), the output lattice is guaranteed to have the same size and resolution as the input lattice. Therefore, a CNN comprising purely convolution layers, without any

[1] Note that the definition of receptive fields using the notion of Minkowski sum may be applicable only to our particular setting, in which we are considering both the input and output vector fields over the same domain $\mathbb{R}^2$.

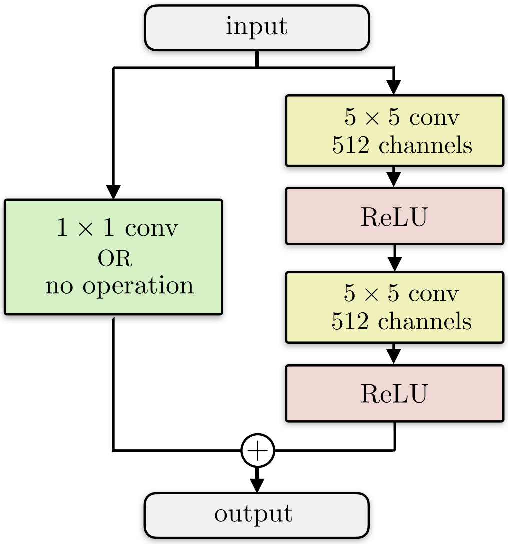

FIG. 2. Diagram of a ResNet block. The input to a Resnet block goes through two different pathways: the skip connection, where no operation is performed if input and output have the same number of channels or a 1 × 1 convolution layer otherwise, and normal connection, where two 5 × 5 convolution-ReLU activation blocks are stacked on top of each other. The results of the two pathways are then added together as the output.

fully connected layers, can be scaled to an arbitrary lattice of practically any size, as long as the lattice element has locally the same geometric and topological structures.

Meanwhile, the architecture of the ResNet block in the backbone is described in Fig. 2. Similar to the original ResNet, the input goes through two separate pathways. One pathway (*right* path in Fig. 2) comprises two convolution layers with the ReLU activation [42], stacked on top of each other, in order to develop a feature vector characterizing patterns of the input vector field at each local neighborhood on the pixel grid. The other pathway (*left* path in Fig. 2) can be either the skip connection, in which the input values are directly copied without any transformation, or a 1 × 1 convolution, in which the input feature vector at each grid location undergoes a dimensionality reduction. The outputs from the two pathways are then added together to produce the overall output of the ResNet block. Note that we do not employ the batch normalization technique, which is a technique used in the original ResNet model to avoid the vanishing gradient problem. Empirically, we found that batch normalization overly regularized the network causing severe underfitting of the spin torque and overall deteriorating the prediction performance. Moreover, since the input spin vectors are already well normalized to have a length of 1, it is not necessary to employ batch normalization.

C. Training

Our training and testing sets consist of 60 independent spin dynamics simulations, respectively, performed on a 48 × 48 triangular lattice. The following parameters are used for the *s-d* Hamiltonian (4): The nearest-neighbor hopping was set to $t_1 = 1$, which also provides the reference unit for energy. A third-neighbor hopping $t_3 = -0.85$ is included in order to stabilize a triple-Q magnetic order that underlies the skyrmion lattice (SkL) phase [44]. The electron-spin coupling constant is set at $J = 1$. An electron chemical potential $\mu = -3.5$ was used, and an external magnetic field $H_{\rm ext} = 0.005$ was included to explicitly break the time-reversal symmetry and induce the SkL [44]. As discussed in Sec. II, the exchange fields $\mathbf{H}_i$ acting on spins are obtained by solving the electron Hamiltonian. Specifically, using Eq. (3) and the *s-d* Hamiltonian (4), the exchange fields are given by

$$\mathbf{H}_i = J \sum_{\alpha,\beta=\uparrow,\downarrow} \boldsymbol{\sigma}_{\alpha\beta}\, \rho_{i\beta,i\alpha}, \tag{10}$$

where $\rho_{i\alpha,j\beta} := \langle c^\dagger_{j\beta} c_{i\alpha} \rangle$ is the electron correlation function, or single-electron density matrix. The kernel polynomial method (KPM) [45,46] was used to compute the electron density matrix for generating the training dataset. The KPM is more efficient compared with exact diagonalization, yet is considered numerically exact when a large number of Chebyshev polynomials and random vectors are used.

The timescale of the precession dynamics of the LLG equation (1) is given by $t_0 = (\gamma J S)^{-1}$, where γ is the gyromagnetic ratio, J is the electron-spin coupling, and S is the length of the localized magnetic moments. The damping term introduces another timescale $t_{\rm damping} = t_0/\alpha$ which characterizes the rate of energy dissipation, where α is a dimensionless coefficient. In the following, the simulation time is measured in terms of t_0, and a damping coefficient $\alpha = 0.05$ is used.

The initial conditions of the simulations are divided into two types. The first one is perturbed SkL, where a periodic array of skyrmions is baked into the initial condition but spins had random noise added. The other type is random initialization where the spins are totally randomly generated. For each type of initial condition, a total of 30 simulations were generated, each of them comprised of 5000 time steps. For a given initial condition, a semi-implicit second-order scheme [47] which preserves the spin length was employed to integrate the LLG equation (1) with a time step $\Delta t = 0.1$.

The spins and their corresponding exchange fields at all lattice sites were collected every 10 other steps in the simulation. We focused on the training of the electron-induced exchange field, so the external constant field of $H_{\rm ext} = 0.005$ in the z direction was removed. The field $\mathbf{H}_i$ was then decomposed into components that are parallel and perpendicular to spin components, and only the perpendicular component, which is equivalent to the torque $\mathbf{T}_i$, was kept as the parallel component has no effect in the evolution of spin configuration and is around two orders of magnitude larger than the perpendicular component. The perpendicular fields were then normalized to have a mean magnitude of 1 over the entire dataset. Then 70% of the entire dataset was used for the training, while the rest was set aside for validation. The split of the dataset is stratified so the training and testing set has the same proportion of the two types of simulations.

The triangular-lattice *s-d* Hamiltonian in Eq. (4) is invariant under two independent symmetry groups: the SO(3)/SU(2) rotation of spins and the D_6 point group of the triangular lattice. Here the rotation symmetry refers to the global rotation of local magnetic moments $\mathbf{S}_i \to \mathcal{R} \cdot \mathbf{S}_i$

(treated as classical vectors), and a simultaneous unitary transformation of the electron spinor $\hat{c}_{i\alpha} \rightarrow \hat{U}_{\alpha\beta}\hat{c}_{i\beta}$, where $\mathcal{R}$ is an orthogonal 3×3 matrix and $\hat{U} = \hat{U}(\mathcal{R})$ is the corresponding 2×2 unitary rotation operator. The ML model, corresponding to an effective force-field model by integrating out electrons, needs to preserve the SO(3) rotation symmetry of spin, which means under uniform rotation $\mathcal{R}$ of all spins in the neighborhood, the ML predicted spin torques should undergo the same rotation transformation $\mathbf{T}_i \rightarrow \mathcal{R} \cdot \mathbf{T}_i$. On the other hand, under a symmetry operation g of the D_6 point group centered at some lattice point, both spins and torques transform under the D_6 point group as $\mathbf{S}_i \rightarrow \mathbf{S}_j$ and $\mathbf{T}_i \rightarrow \mathbf{T}_j$, where the lattice points $\mathbf{r}_j = \mathcal{R}(g) \cdot \mathbf{r}_i$, and $\mathcal{R}(g)$ denotes the 3×3 matrix corresponding to g.

To incorporate both symmetries into the CNN model, we introduced data augmentation during our training phase. Specifically, for each input spin configuration and the corresponding torque field, a random SO(3) rotation field was applied to the spins $\{\mathbf{S}_i\}$ and a random D_6 symmetry operation was applied to the lattice points. The same symmetry operations, both for spin-space and real-space lattice, were also applied to the torque fields $\{\mathbf{T}_i\}$. These additional symmetry-generated input/output configurations were included along with the original ones to the dataset for supervised training. We note that, contrary to previous ML models where the symmetry is explicitly included through descriptors, the symmetry of the itinerant electron Hamiltonian is enforced on the ML vector model statistically in this data-driven approach.

Since the torques in the dataset could differ at most by one order of magnitude, we find that the usual mean absolute error or mean square error loss functions do not perform very well. Instead, we adopted a mean percentage absolute loss:

$$L = \frac{1}{N}\sum_{i=1}^{N} \frac{\left|T_i^x - \hat{T}_i^x\right| + \left|T_i^y - \hat{T}_i^y\right| + \left|T_i^z - \hat{T}_i^z\right|}{\left|\hat{\mathbf{T}}_i\right|}, \tag{11}$$

where N is the total number of lattice sites within each batch summed across all lattices, $\hat{\mathbf{T}}_i$ is the ground truth field vector at ith lattice site and $\mathbf{T}_i = (T_i^x, T_i^y, T_i^z)$ is the predicted field vector and its three components.

An Adam [48] optimizer with an initial learning rate of 10^{-3} was used for the training. The learning rate was later reduced to 10^{-6} upon the plateau of loss value in the testing set. We did not use any regularization methods, such as dropout or weight decay, and there is no evidence of overfitting when comparing training and testing set loss values. The model and its training process are implemented in PyTorch [49], and training was performed on one Nvidia A100 for roughly 72 hours.

IV. RESULTS

Here we present benchmarks of the CNN models by comparing spin torque predictions and small-scale dynamical simulations against exact methods. We further demonstrate the restoration and stability of a skyrmion lattice in large-scale LLG simulations, highlighting the scalability and transferability of our ML approach.

A. Benchmark of spin torque prediction

The spin torques $\mathbf{T}_i$ predicted from the trained CNN model are compared against the ground truth in Fig. 3 using configurations from the test dataset. Two types of testing data are used for this benchmark: LLG simulations of an initially perturbed SkL state, and LLG simulations starting from random spins. In both cases, the predicted torque components closely follow the ground truth with roughly equal variance across the entire range. Note the values of torque components in the random spin case span a range nearly twice larger than that of the SkL case. As can be expected, the ML model performs better in the case of the SkL simulations since spin configurations here correspond to a rather small and special set of the whole state space. Yet a fairly good agreement was obtained even for the testing dataset with completely random initial spins.

We further examine the magnitude of predicted torques versus the ground truth, as well as the angle between the predicted field vector and ground truth vector in Fig. 4. Again, an overall satisfactory agreement was obtained, with the majority of the predictions close to or symmetrically distributed around the ground truth value. Note that due to the distortion of the logarithmic function, the same deviation from ground truth at large and small magnitudes will look asymmetric and "biased" towards smaller values. Therefore, two red dotted lines with constant deviation of 10^{-2} (outer) and 10^{-3} (inner) have been added in Fig. 4(a). Even at a large magnitude where the error of the ML model is also the largest, the difference in field vector magnitude is almost guaranteed to be smaller than 10^{-2}. At a small magnitude, the difference in field vector magnitude is most likely to be smaller than 4×10^{-3} and would typically be around 10^{-3}. We did not notice any bias in our ML prediction results. The ML-predicted vectors are also very closely aligned with ground truth field vectors. As shown in Fig. 4(b), most vectors would have an angle smaller than 10°, and it is almost impossible to find a predicted vector with a more than 30 degree angle from its ground truth counterpart.

B. Dynamical benchmark

In addition to accurate predictions of spin torques, another important benchmark is whether the trained ML model can also faithfully capture the dynamical evolution of the itinerant spin model. To this end, we integrated the trained CNN model into the LLG dynamics simulations and compared the results with LLG simulations based on KPM [45,46]. We consider simulations of a thermal quench process where an initially random magnet is suddenly quenched to nearly zero temperature at time $t = 0$. While our trained CNN model produces fairly accurate spin torques, small prediction errors still persist, as discussed in the previous section. Statistically, these prediction errors are similar to the stochastic noise $\boldsymbol{\tau}_i(t)$ in the LLG equation (1). These site-dependent fluctuating random torques are similar to the thermal forces in Langevin dynamics. Both random forces are physically due to thermal fluctuations through coupling to a thermal bath at a fixed temperature. As a result, while the temperature of the ML-LLG simulations was set to exactly zero, a very low yet nonzero temperature $T = 0.001$ was introduced in the exact LLG dynamics to mimic the prediction error.

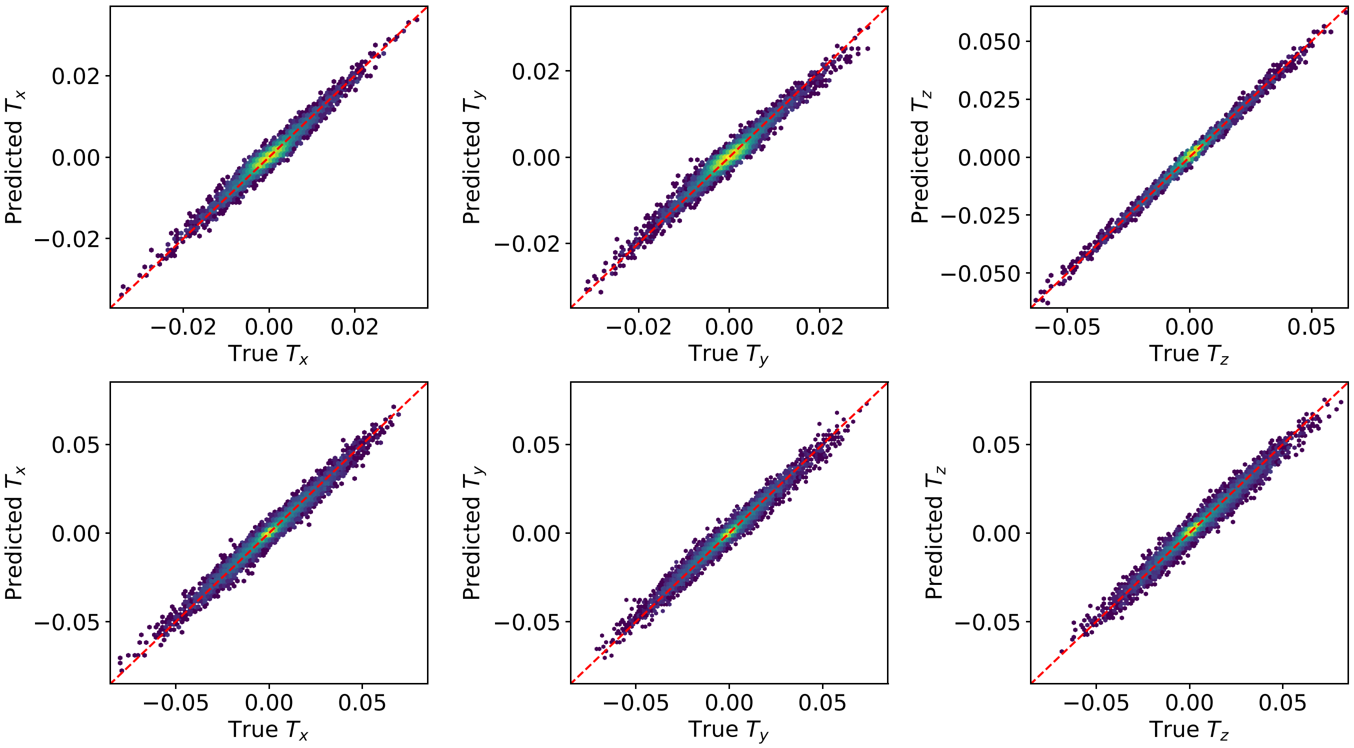

FIG. 3. Predicted spin torque components (T_x, T_y, T_z) vs ground truth components from the testing set. The red-dotted diagonal lines indicate perfect prediction. The top row shows the prediction results based on spin configurations obtained from LLG simulations of a perturbed SkL. Results from LLG simulations with random initial states are shown in the bottom row.

The model parameters of the s-d Hamiltonian are chosen to stabilize a spontaneous SkL ground state. Importantly, the emergence of skyrmion crystal not only breaks the spin-rotation symmetry but also breaks the lattice translational symmetry. The periods of this spatial modulation, i.e., the lattice constant of the skyrmion lattice, are determined by the underlying electron Fermi surface. Indeed, while an SkL state can be intuitively thought of as a periodic array of particle-like spin textures, physically SkL phases often result from an instability caused by quasinesting of the electron Fermi surface that gives rise to a multiple-Q magnetic order [44,50,51].

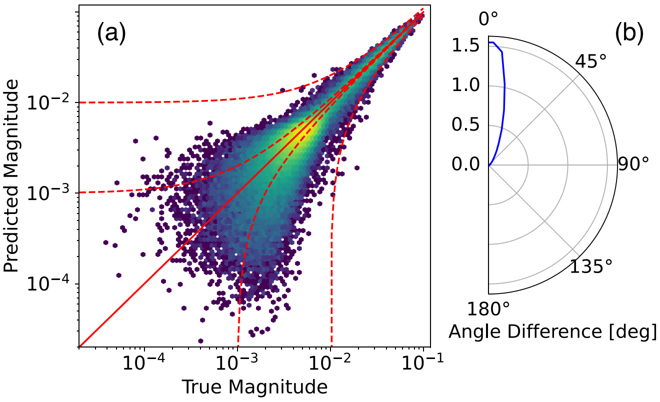

FIG. 4. (a) Comparison of predicted field vector magnitude against ground truth. The red line indicates prediction equaling to ground truth, the outer red dotted line represents a 10^{-2} deviation from ground truth magnitude while the inner one represents a deviation of 10^{-3}. The color denotes the log density. (b) Angular difference between ground truth field vector and predicted field vector.

In our case, the geometry of the Fermi surface at the chemical potential $\mu = -3.5$ allows significant segments to be connected by three wave vectors $\mathbf{Q}_1 = (\pi/3a, 0)$ and $\mathbf{Q}_{2,3} = \mathcal{R}_{\pm 2\pi/3} \cdot \mathbf{Q}_1$, related to each other by symmetry operations of the D_6 group. Here a is the lattice constant of the underlying triangular lattice. This means that maximum energy gain through electron-spin coupling is realized by spin helical orders with one of the above three wave vectors. Further analysis shows that the electron energy is further lowered by the simultaneous ordering of all three wave vectors, giving rise to an emergent triangular lattice of skyrmions.

The relaxation of the magnet after the thermal quench is dominated by the formation of the triangular SkL. A perfect SkL is distinguished by six Bragg peaks at $\mathbf{q} = \pm\mathbf{Q}_1$, $\pm\mathbf{Q}_2$, and $\pm\mathbf{Q}_3$ in momentum space. Yet, since the spin interactions are local in nature, the crystallization of skyrmions is inherently an incoherent process. Small crystallites of skyrmions are nucleated randomly separated by large domains of disjointed structures. To quantitatively characterize this crystallization process, we compute the time-dependent spin structure factor, which is defined as the square of the Fourier transform of the spin field

$$\mathcal{S}(\mathbf{q}, t) = \left\langle \left| \frac{1}{N} \sum_{i=1}^{N} \mathbf{S}_i(t) \exp(i\mathbf{q} \cdot \mathbf{r}_i) \right|^2 \right\rangle, \tag{12}$$

where the bracket $\langle \cdots \rangle$ indicates averaging over thermal ensemble as well as initial conditions. The structure factor is

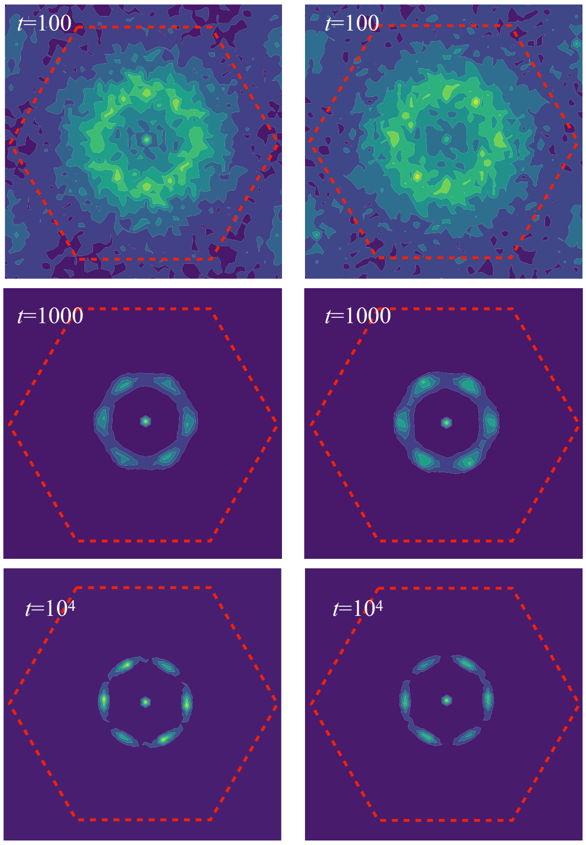

FIG. 5. A comparison of spin structural factors obtained by averaging 30 independent LLG simulations based on KPM (left) and ML model (right). The same set of random initial conditions on a 48 × 48 triangular lattice were used in both simulations. The red dashed lines indicate the first Brillouin zone of the momentum space.

itself the Fourier transform of the spin-spin correlation function in real space and can be directly measured in neutron scattering experiments. The spin structure factors at various times after the quench, obtained from LLG simulations based on both KPM and ML models, are shown in Fig. 5. Due to the stochastic nature of such simulations, the results are obtained by averaging 30 independent runs. Overall, the results from LLG simulations with the trained CNN model agree very well with those based on the numerically exact KPM.

Both simulations show that a ringlike structure quickly emerges in the structure factor after the quench. The radius of the ring is close to the length of the three nesting wave vectors $\mathbf{Q}_\eta$, indicating the initial formation of skyrmions. As the system relaxed towards equilibrium, the ringlike structure becomes sharper. Moreover, the spectral weight starts to accumulate at the six spots corresponding to the $\pm\mathbf{Q}_\eta$ wave vectors. Physically, the emergence of the six broad segments corresponds to the growth of domains of the skyrmion lattice. The size of these intermediate skyrmion crystallites can be inferred from the width of the six spots. However, both simulations found that even at a late stage of the equilibration, the structure factor exhibits only six diffusive peaks at the nesting wave vectors, instead of sharp Bragg peaks as expected for a perfect SkL. The broad peaks at a late stage of the phase ordering thus indicate an arrested growth of SkL domains in real space. An example of the real-space spin configuration at $t = 10^4$ after the quench is shown in Fig. 6. The snapshot shows rather small triangular clusters of skyrmions coexist with stripelike structures of different orientations, These stripes or helical spins corresponds to the single-Q magnetic order which are metastable states of the s-d model.

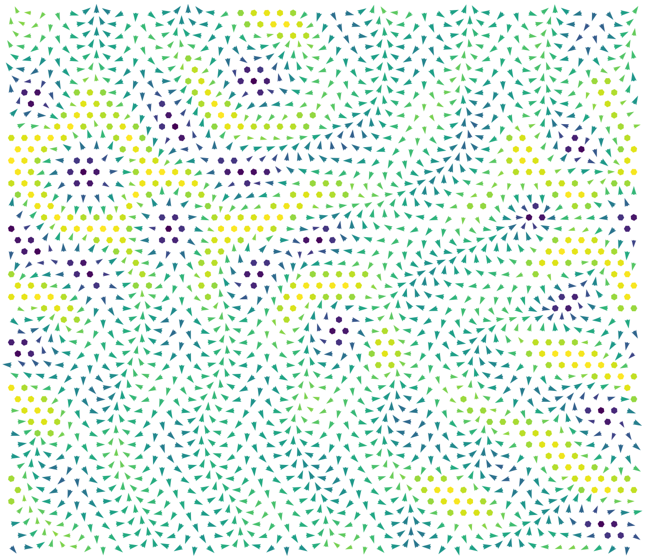

FIG. 6. Snapshot of the spin configuration at the end of the LLG simulation with random initial conditions on a 48 × 48 triangular lattice.

This intriguing freezing phenomenon can be partly attributed to the frustrated electron-mediated spin interactions. Another important source is related to the degeneracy between skyrmions of opposite vorticity, or circulation of the in-plane spins. The two opposite circulations correspond to the topological winding number $w = \pm 1$ for the skyrmions. As discussed above, the spin-rotation symmetry is decoupled from the lattice in the s-d Hamiltonian (4), which provides a minimum model for centrosymmetric itinerant magnets without spin-orbit coupling. As a result, skyrmions with clockwise circulation is energetically degenerate with counterclockwise ones. This also means that SkL domains of the two opposite circulations are nucleated with roughly the same probability after the thermal quench. The subsequent annihilation of skyrmions with opposite vorticity thus prohibits the growth of a large coherent SkL.

C. Scalability and large-scale simulation

As discussed in Sec. III B, due to the locality property and the fixed-size kernels, the CNN model can be directly scaled to larger lattice systems without retraining, thus enabling large-scale dynamical simulations that are beyond conventional approaches. Here we demonstrate the scalability of the CNN spin-torque model by applying it to LLG simulations of large-scale SkL phases. Specifically, we perform LLG simulations of a perturbed SkL state on a 96 × 96 lattice using a CNN model trained from simulations of a 48 × 48 lattice. As discussed above, the triangular skyrmion lattice, characterized by the three nesting wave vectors, can be viewed as a superposition of three helical spin orders. Explicitly, a perfect SkL

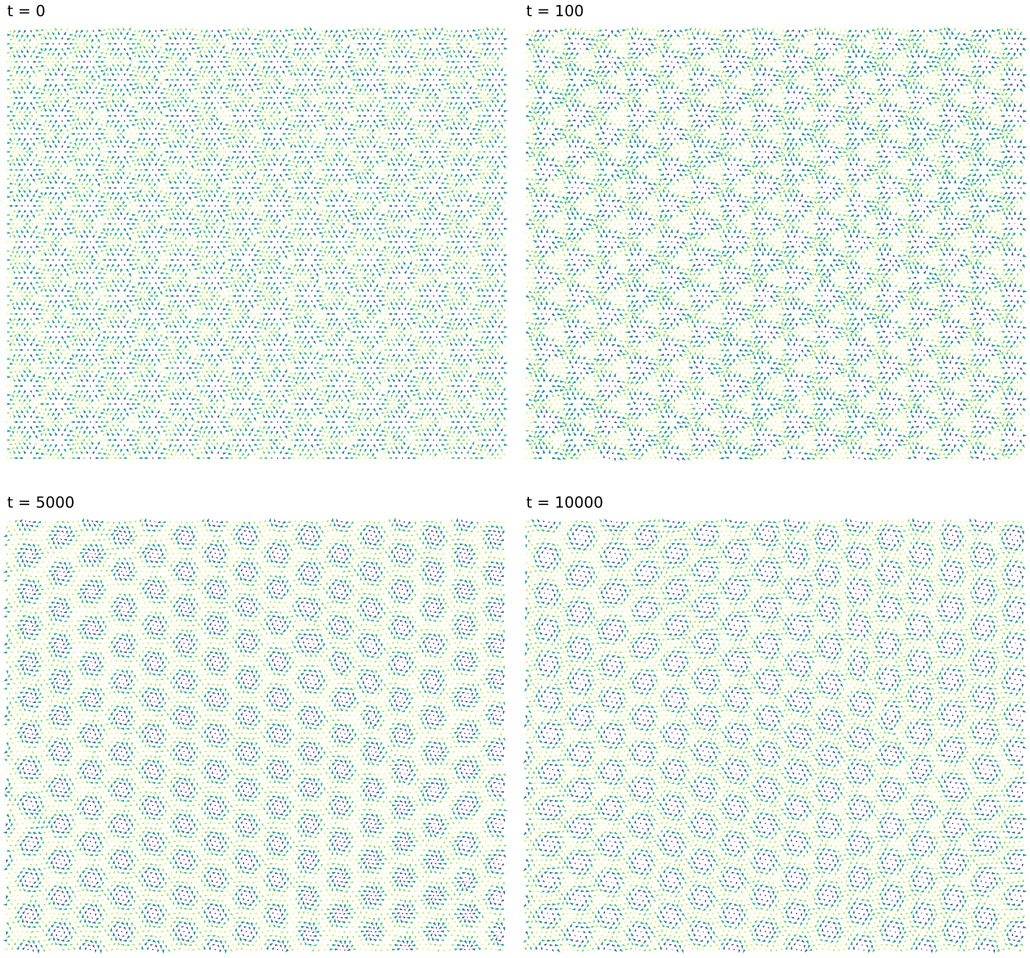

FIG. 7. CNN-based LLG simulation on a 96 × 96 lattice showing the restoration of a perturbed SkL. The CNN model was originally trained on a 48 × 48 lattice. The initial spin configuration is given by the SkL ansatz (13) with additional site-dependent random phases and amplitudes of S_z.

can be approximated by the following ansatz [44,51]:

$$\begin{aligned}\mathbf{S}_i \sim \left(\cos \mathcal{Q}_{1i} - \frac{1}{2}\cos \mathcal{Q}_{2i} - \frac{1}{2}\mathcal{Q}_{3i}\right)\hat{\mathbf{e}}_1 \\ + \left(\frac{\sqrt{3}}{2}\cos \mathcal{Q}_{2i} - \frac{\sqrt{3}}{2}\cos \mathcal{Q}_{3i}\right)\hat{\mathbf{e}}_2 \\ + \left[A\left(\sin \mathcal{Q}'_{1i} + \sin \mathcal{Q}'_{2i} + \sin \mathcal{Q}'_{3i}\right) + M\right]\hat{\mathbf{e}}_3,\end{aligned} \tag{13}$$

where $\hat{\mathbf{e}}_{1,2,3}$ are three orthogonal unit vectors, $\mathcal{Q}_{\eta i} = \mathbf{Q}_\eta \cdot \mathbf{r}_i$, $\mathcal{Q}'_{\eta,i} = \mathcal{Q}_{\eta,i} + \phi$ are phase factors of the three helical orders, and ϕ, A, and M are fitting parameters. To demonstrate that the ML model can indeed stabilize the SkL, which is the ground state of our chosen *s*-*d* Hamiltonian, we initialize the system with a perturbed array of skyrmions as shown in Fig. 7(a). The randomness in the initial state was introduced by allowing site-dependent parameters ϕ_i, A_i and m_i, which are randomly generated, in the above SkL ansatz (13). Contrary to completely random spins for the initial states in the previous dynamical benchmark, this initial state preserves a coherent structure of skyrmion winding numbers. As these topological numbers have to be conserved, the relaxation of the system is free of random annihilation of skyrmions. As shown in Fig. 7, our ML-based LLG simulations indeed find that a nearly perfect SkL is restored and stabilized over a long period of simulation time.

We further investigate the scalability in the time domain by running our ML-based LLG simulation long past the duration of the training simulations. Figure 8 shows a roughly constant structural factor long beyond the duration of simulation snapshots used in training. While we noticed a huge decrease in structural factor between the time of 15 000 and 23 000, the structural factor quickly rebounds to its original stable value [$\mathcal{S}(\mathbf{q}, t) \approx 305$]. These temporal fluctuations can be ascribed to the prediction errors of the ML model. Yet, as also discussed above, such errors play a role similar to the stochastic noise in Langevin-type dynamics simulations. Our results thus demonstrate the robustness of SkL under small random perturbations. Importantly, this further benchmark highlight the scalability of our ML models not only in spatial domains (larger lattices), but also in temporal scales (much longer simulation times) as well.

D. Symmetry requirements

In order to incorporate the underlying symmetries of a physical system into an ML model, one needs to introduce appropriate biases (prior knowledge) through the statistical

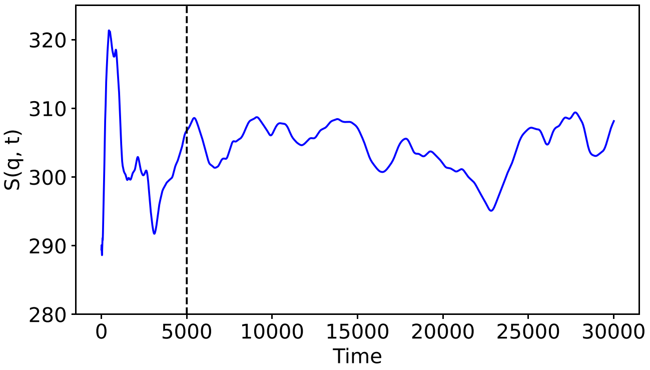

FIG. 8. The evolution of structural factor of a much longer than training simulation with time, using perturbed Skyrmion initial condition. The dashed black line indicates the duration of the training simulation, and our ML-based LLG simulation is capable of keeping the structural factor constant for more than five times the duration of the training simulation.

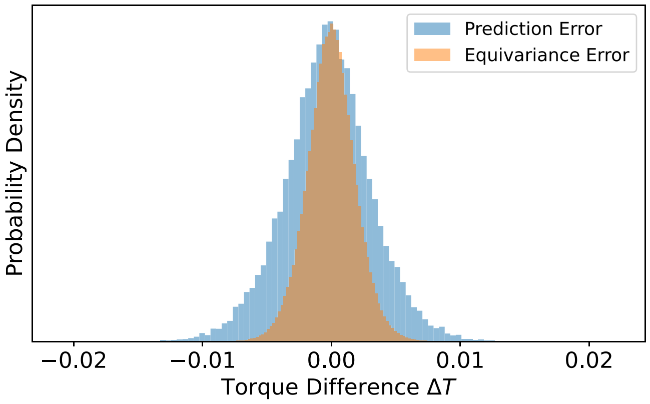

FIG. 9. Distribution of the equivariance error $err_{eq} := f_{\text{CNN}}(\{R\mathbf{S}_i\}) - Rf_{\text{CNN}}(\{\mathbf{S}_i\})$ where R is an arbitrary rotation. The overall prediction error ("predicted torque" minus "ground truth torque") on the test dataset is superimposed for comparison. The equivariance error is sufficiently smaller than the overall prediction error, implying that the model can preserve the underlying symmetry of the physics system reasonably well.

learning process. Two of the major approaches to this end are (i) *data augmentation* based on the symmetry group of the system, (ii-a) constructing *symmetry-invariant* descriptors, or (ii-b) constructing *equivariant* neural network architectures w.r.t. the symmetry group. These two types of approaches correspond to introducing the *observational* and the *inductive* biases, respectively, in the context of physics-informed machine learning literature (see, e.g., [52–57]). As discussed in Sec. III C, the *local* symmetries of our system, i.e., the spin-space and the real-space lattice symmetries, a.k.a. the internal (gauge) and the spacetime symmetries [58], consist of G-valued fields over the underlying lattice, where $G = \text{SO}(3) \times \text{D}_6$. In the present work, we adopted the data augmentation approach as the mean to enforce the symmetry constraint for the reasons justified as follows.

First, we briefly summarize the theoretical justification of how data augmentation during our training phase is injecting the above-mentioned symmetries into the underlying supervised learning process (see [54,59,60] for details). To avoid cumbersome notation, we denote a pair of spin configurations and its corresponding torque field $(\mathbf{S}, \mathbf{T}) = (\{\mathbf{S}_i\},\ \{\mathbf{T}_i\})$ by $\mathbf{F}$. Our training data $\mathbf{F}_1, \dots, \mathbf{F}_n$ consist of independent identically distributed (i.i.d.) samples from a probability distribution $\mathbb{P}$ over the space of all spin-torque fields. It is of fundamental importance that the probability distribution $\mathbb{P}$ remains *invariant* under the *action* of each local symmetry $g \in \mathcal{G}$, where $\mathcal{G}$ denotes the space of all local symmetries of the system.[2] Hence, the data augmentation process can be considered as enriching our set of samples from the probability distribution $\mathbb{P}$, where our goal is to learn it, by adding transformed spin-torque fields $g \cdot \mathbf{F},\ g \in \mathcal{G}$.

During the training procedure, at each step t, a minibatch B_t of spin-torque $(\mathbf{S}, \mathbf{T})$ samples of size $|B_t|$ is chosen, and a random local symmetry $g_{t,b} \in \mathcal{G}$ is applied to each spin $\mathbf{S}_b$ and torque $\mathbf{T}_b$ field, $b \in B_t$. Then, according to the stochastic gradient descent (SGD) algorithm, the parameters θ of the CNN model f_θ get updated as

$$\theta_{t+1} = \theta_t - \frac{\eta}{|B_t|} \sum_{b \in B_t} \nabla_\theta L(f_\theta(g_{t,b} \cdot \mathbf{S}_b),\ (g_{t,b} \cdot \hat{\mathbf{T}}_b)), \tag{14}$$

where L denotes the loss function given by Eq. (11), and η is the learning rate. In other words, the augmented SGD can be considered as minimization of the *empirical risk* associated with the following *augmented loss function:*

$$\int_{\mathcal{G}} L(f_\theta(g \cdot \mathbf{S}),\ (g \cdot \hat{\mathbf{T}}))\, d\mathbb{Q}(g) \tag{15}$$

in which one takes an average along the whole orbit of the group action w.r.t. a probability distribution $\mathbb{Q}$ over $\mathcal{G}$. It can be proved that data augmentation based on the underlying symmetry group reduces the variance of general estimators and improves their generalizability [54].

The above theoretical justification can further be validated through empirical data. Figure 9 shows the typical prediction error (blue), the difference between predicted and ground truth torque, and the equivariance error (orange), defined as $f_\theta(g \cdot \mathbf{S}) - g \cdot f_\theta(\mathbf{S})$. As can be seen in the figure, the equivariance error is smaller than the typical prediction error, indicating that in practice data augmentation employed is capable of preserving the underlying symmetry of the physics system to a satisfactory degree.

E. Locality principle and receptive fields

To attest to the locality principle, we analyze the receptive field of our CNN model in this section. As discussed in Sec. III A, the receptive field of a convolution layer f_m is defined to be the support $\text{supp}(h_m)$ of the corresponding convolution kernel h_m, i.e., the region where the function values of h_m are nonzero tensors. The receptive field of the

[2] The action of a local symmetry $g \in \mathcal{G}$ over a spin $\mathbf{S}$ and a torque $\mathbf{T}$ field is the induced transformation by g. We denote it by $g \cdot \mathbf{F} := (g \cdot \mathbf{S},\ g \cdot \mathbf{T})$.

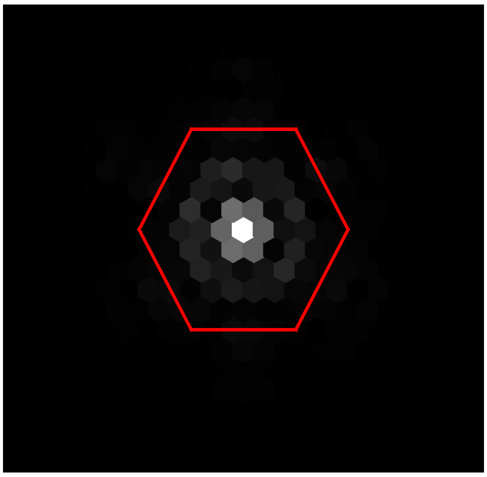

FIG. 10. The effective receptive fields (ERFs) of the ML model. Since for each lattice site both the input and output tensors have three components, there are in total nine ERFs corresponding to the nine partial derivatives of the three output torque components with respect to the three input spin components. The sum of these absolute values of these derivatives are then presented in this figure, with blacker pixels indicating smaller derivative value. The red line roughly traces the nonzero value regions of the ERF.

entire CNN model is computed as the Minkowski sum of the receptive fields of individual convolution layers, or RF = $\mathrm{supp}(h_1) \oplus \cdots \oplus \mathrm{supp}(h_L)$. For our model, in which there are 10 layers in depth with each layer comprising 5×5 convolution kernels of stride 1, the size of the receptive field is calculated to be 41. This implies that, in principle, the spin directions of the 41-neighborhood can influence the prediction of the torque at the lattice position i.

However, the naïve computation of the receptive field size may be misleading because the kernel size of a convolution layer simply just indicates the theoretical maximum of the receptive field. On the other hand, the actual region of nonzero values can be much smaller than the theoretical receptive field size. To this end, we used the approach of Luo *et al.* [61] to compute the *effective* receptive field size, in which function values are practically nonzero. Figure 10 shows the result of such a calculation performed on the trained CNN model. The red hexagonal line delineates the region inside the hexagon where function values are practically nonzero and the region outside the hexagon where function values are practically zero. The grayscale values inside the hexagon indicate different levels of influence of neighboring spins in computing the torque vector. As can be seen from the figure, at the lattice location i, which is the center of the red hexagon, the weighting factor is the largest, implying that $\mathbf{T}_i$ is predominantly determined by $\mathbf{S}_i$. The 1-neighborhood $\mathcal{N}_1(i)$, i.e., the immediate neighbors to the lattice location i, also has bright intensity values, implying that the relative configuration of the spin direction $\mathbf{S}_i$ to its neighboring spins $\mathbf{S}_j$ at $j \in \mathcal{N}_1(i)$ also have a significant influence to the output torque $\mathbf{T}_i$. Similarly, it appears that the spin directions of the 3-neighborhood have strong influences on torque prediction, while small influence can be detected all the way to 6-neighborhood.

This result is consistent with previous ML spin-torque models based on symmetry-invariant descriptors [33,34], which shows that the spin dynamics of similar *s*-*d* models can be nicely captured by BP-type models based on fully connected NN with input from a neighborhood up to $r_c \sim 5$ lattice constants. Physically, as discussed above, the finite sizes of effective receptive field are due to the locality nature of the spin torques. However, the range of locality can only be indirectly determined from exact calculations. In practice, the cutoff radius is treated as an ad hoc parameter in BP-type ML models, or is determined through trial and error. It is thus worth noting that the CNN model offers a systematic and rigorous method to determine this important physical attribute of electronic models.

V. CONCLUSION AND OUTLOOK

In this paper, we presented a CNN model to predict spin torques directly from input spin configuration for large-scale LLG dynamics simulations of itinerant magnets. Our CNN model is purely convolutional without any fully connected (dense) layers, and thus presents a distinct advantage of built-in locality. Central to each CNN layer is the convolution with a kernel or filter, which can be viewed as a Green's function representing finite responses to a local source. As each kernel is characterized by a finite set of trainable parameters, the CNN model can be used for dynamical simulations on larger system sizes without rebuilding or retraining the neural network. We demonstrated our ML approach on a triangular-lattice *s*-*d* model which exhibits a skyrmion crystal in its ground state. Using the ML-predicted torques in the LLG dynamics simulations, we showed that the trained CNN model can successfully reproduce the relaxation of the skyrmion phase of the itinerant spin models. We further demonstrated the scalability and transferability of our approach by showing that large-scale LLG simulations based on our CNN model are able to stabilize a perturbed skyrmion lattice and maintain it for a long period of time.

Contrary to the ML force-field models based on the Behler-Parrinello scheme, our CNN model directly predicts torques, which are the spin analog of atomic forces. In BP-type approaches, ML models, either Gaussian process regression or fully connected neural nets, are built to predict local energy, which cannot be directly compared with exact calculations. The forces are obtained from derivatives of the total energy, which is the sum of all local energies. The introduction of local energy takes advantage of the locality property and also facilitates the incorporation of symmetry into the ML models. Yet, the fact that forces are computed indirectly from derivatives of energy also restricts BP-type models to the representation of conservative forces and quasi-equilibrium electron systems. On the other hand, our CNN approach can be used to describe both conservative as well as nonconservative spin torques. This capability is particularly important for ML modeling of out-of-equilibrium driven systems where the electron-mediated torques are nonconservative. A representative example is the spin transfer torque which plays an important role in spintronics applications.

For future work, we are currently looking into ways to enforce constraints due to either symmetry or conservation laws more strictly and rigorously. To this end, previous computer vision literature on equivariant CNNs (see, e.g., Geiger and Smidt [62]) may shed light on how to constrain CNN layers to preserve SO(3) and D_6 symmetries. Moreover, the present work is limited to approximating torques using spin directions at each time step and does not provide a direct solution to the LLG equation in Eq. (1). In a recent body of literature, however, there have been attempts to solve governing partial differential equations (PDE) of physics directly using so-called physics-aware deep neural networks (see, e.g., Nguyen *et al.* [55]). Using these physics-aware CNN methods, we expect to attain faster and more accurate approximations of the spin dynamics, which is going to be another meaningful direction of research.

ACKNOWLEDGMENTS

S.Z. and G.-W.C. acknowledge the support from the U.S. Department of Energy Basic Energy Sciences under Contract No. SC0020330. S.S.B. and P.C.H.N. were supported by the National Science Foundation under Grant No. 2203580. X.C. acknowledges the support from the Jefferson Scholars Foundation and the benefactor of the Edward P. Owens Jefferson Fellow.

[1] A. N. Bogdanov and D. A. Yablonskii, Thermodynamically stable “vortices” in magnetically ordered crystals: The mixed state of magnets, Sov. Phys. JETP **68**, 101 (1989).

[2] U. K. Rößler, A. N. Bogdanov, and C. Pfleiderer, Spontaneous skyrmion ground states in magnetic metals, Nature (London) **442**, 797 (2006).

[3] S. Mühlbauer, B. Binz, F. Jonietz, C. Pfleiderer, A. Rosch, A. Neubauer, R. Georgii, and P. Böni, Skyrmion lattice in a chiral magnet, Science **323**, 915 (2009).

[4] X. Z. Yu, Y. Onose, N. Kanazawa, J. H. Park, J. H. Han, Y. Matsui, N. Nagaosa, and Y. Tokura, Real-space observation of a two-dimensional skyrmion crystal, Nature (London) **465**, 901 (2010).

[5] X. Z. Yu, N. Kanazawa, Y. Onose, K. Kimoto, W. Z. Zhang, S. Ishiwata, Y. Matsui, and Y. Tokura, Near room-temperature formation of a skyrmion crystal in thin-films of the helimagnet FeGe, Nat. Mater. **10**, 106 (2011).

[6] S. Seki, X. Z. Yu, S. Ishiwata, and Y. Tokura, Observation of skyrmions in a multiferroic material, Science **336**, 198 (2012).

[7] N. Nagaosa and Y. Tokura, Topological properties and dynamics of magnetic skyrmions, Nat. Nanotechnol. **8**, 899 (2013).

[8] D. Loss and P. M. Goldbart, Persistent currents from Berry’s phase in mesoscopic systems, Phys. Rev. B **45**, 13544 (1992).

[9] J. Ye, Y. B. Kim, A. J. Millis, B. I. Shraiman, P. Majumdar, and Z. Tešanović, Berry Phase Theory of the Anomalous Hall Effect: Application to Colossal Magnetoresistance Manganites, Phys. Rev. Lett. **83**, 3737 (1999).

[10] G. Tatara and H. Kawamura, Chirality-driven anomalous hall effect in weak coupling regime, J. Phys. Soc. Jpn. **71**, 2613 (2002).

[11] M. V. Berry, Quantal phase factors accompanying adiabatic changes, Proc. R. Soc. London A **392**, 45 (1984).

[12] R. F. L. Evans, W. J. Fan, P. Chureemart, T. A. Ostler, M. O. A. Ellis, and R. W. Chantrell, Atomistic spin model simulations of magnetic nanomaterials, J. Phys.: Condens. Matter **26**, 103202 (2014).

[13] A. Vansteenkiste, J. Leliaert, M. Dvornik, M. Helsen, F. Garcia-Sanchez, and B. Van Waeyenberge, The design and verification of MuMax3, AIP Adv. **4**, 107133 (2014).

[14] D. Marx and J. Hutter, *Ab Initio Molecular Dynamics: Basic Theory and Advanced Methods* (Cambridge University Press, Cambridge, 2009).

[15] J. Behler and M. Parrinello, Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces, Phys. Rev. Lett. **98**, 146401 (2007).

[16] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, Gaussian Approximation Potentials: The Accuracy of Quantum Mechanics, without the Electrons, Phys. Rev. Lett. **104**, 136403 (2010).

[17] Z. Li, J. R. Kermode, and A. De Vita, Molecular Dynamics with On-the-Fly Machine Learning of Quantum-Mechanical Forces, Phys. Rev. Lett. **114**, 096405 (2015).

[18] V. Botu, R. Batra, J. Chapman, and R. Ramprasad, Machine learning force fields: Construction, validation, and outlook, J. Phys. Chem. C **121**, 511 (2017).

[19] Y. Li, H. Li, F. C. Pickard IV, B. Narayanan, F. G. Sen, M. K. Chan, S. K. Sankaranarayanan, B. R. Brooks, and B. Roux, Machine learning force field parameters from *ab initio* data, J. Chem. Theory Comput. **13**, 4492 (2017).

[20] J. S. Smith, O. Isayev, and A. E. Roitberg, Ani-1: An extensible neural network potential with dft accuracy at force field computational cost, Chem. Sci. **8**, 3192 (2017).

[21] L. Zhang, J. Han, H. Wang, R. Car, and W. E, Deep Potential Molecular Dynamics: A Scalable Model with the Accuracy of Quantum Mechanics, Phys. Rev. Lett. **120**, 143001 (2018).

[22] J. Behler, Perspective: Machine learning potentials for atomistic simulations, J. Chem. Phys. **145**, 170901 (2016).

[23] V. L. Deringer, M. A. Caro, and G. Csányi, Machine learning interatomic potentials as emerging tools for materials science, Adv. Mater. **31**, 1902765 (2019).

[24] R. T. McGibbon, A. G. Taube, A. G. Donchev, K. Siva, F. Hernández, C. Hargus, K.-H. Law, J. L. Klepeis, and D. E. Shaw, Improving the accuracy of Møller-Plesset perturbation theory with neural networks, J. Chem. Phys. **147**, 161725 (2017).

[25] H. Suwa, J. S. Smith, N. Lubbers, C. D. Batista, G.-W. Chern, and K. Barros, Machine learning for molecular dynamics with strongly correlated electrons, Phys. Rev. B **99**, 161107(R) (2019).

[26] T. Mueller, A. Hernandez, and C. Wang, Machine learning for interatomic potential models, J. Chem. Phys. **152**, 050902 (2020).

[27] W. Kohn, Density Functional and Density Matrix Method Scaling Linearly with the Number of Atoms, Phys. Rev. Lett. **76**, 3168 (1996).

[28] E. Prodan and W. Kohn, Nearsightedness of electronic matter, Proc. Natl. Acad. Sci. USA **102**, 11635 (2005).

[29] P. Zhang, S. Zhang, and G.-W. Chern, Descriptors for machine learning model of generalized force field in condensed matter systems, arXiv:2201.00798.

[30] J. Ma, P. Zhang, Y. Tan, A. W. Ghosh, and G.-W. Chern, Machine learning electron correlation in a disordered medium, Phys. Rev. B **99**, 085118 (2019).

[31] Y.-H. Liu, S. Zhang, P. Zhang, T.-K. Lee, and G.-W. Chern, Machine learning predictions for local electronic properties of disordered correlated electron systems, Phys. Rev. B **106**, 035131 (2022).

[32] S. Zhang, P. Zhang, and G.-W. Chern, Anomalous phase separation in a correlated electron system: Machine-learning–enabled large-scale kinetic Monte Carlo simulations, Proc. Natl. Acad. Sci. USA **119**, e2119957119 (2022).

[33] P. Zhang, P. Saha, and G.-W. Chern, Machine learning dynamics of phase separation in correlated electron magnets, arXiv:2006.04205.

[34] P. Zhang and G.-W. Chern, Arrested Phase Separation in Double-Exchange Models: Large-Scale Simulation Enabled by Machine Learning, Phys. Rev. Lett. **127**, 146401 (2021).

[35] P. Zhang and G.-W. Chern, Machine learning nonequilibrium electron forces for spin dynamics of itinerant magnets, npj Comput. Mater. **9**, 32 (2023).

[36] M. A. Brännvall, D. Gambino, R. Armiento, and B. Alling, Machine learning approach for longitudinal spin fluctuation effects in bcc Fe at T_c and under earth-core conditions, Phys. Rev. B **105**, 144417 (2022).

[37] I. Novikov, B. Grabowski, F. Körmann, and A. Shapeev, Magnetic moment tensor potentials for collinear spin-polarized materials reproduce different magnetic states of bcc Fe, npj Comput. Mater. **8**, 13 (2022).

[38] W. F. Brown, Jr., Thermal fluctuations of a single-domain particle, Phys. Rev. **130**, 1677 (1963).

[39] M. A. Ruderman and C. Kittel, Indirect exchange coupling of nuclear magnetic moments by conduction electrons, Phys. Rev. **96**, 99 (1954).

[40] T. Kasuya, A theory of metallic ferro- and antiferromagnetism on Zener's model, Prog. Theor. Phys. **16**, 45 (1956).

[41] K. Yosida, Magnetic properties of Cu-Mn alloys, Phys. Rev. **106**, 893 (1957).

[42] V. Nair and G. E. Hinton, Rectified linear units improve restricted Boltzmann machines, in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)* (Omnipress, Madison WI, 2010), pp. 807–814.

[43] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, New York, 2016), pp. 770–778.

[44] R. Ozawa, S. Hayami, and Y. Motome, Zero-Field Skyrmions with a High Topological Number in Itinerant Magnets, Phys. Rev. Lett. **118**, 147205 (2017).

[45] A. Weiße, G. Wellein, A. Alvermann, and H. Fehske, The kernel polynomial method, Rev. Mod. Phys. **78**, 275 (2006).

[46] Z. Wang, G.-W. Chern, C. D. Batista, and K. Barros, Gradient-based stochastic estimation of the density matrix, J. Chem. Phys. **148**, 094107 (2018).

[47] J. H. Mentink, M. V. Tretyakov, A. Fasolino, M. I. Katsnelson, and T. Rasing, Stable and fast semi-implicit integration of the stochastic Landau–Lifshitz equation, J. Phys.: Condens. Matter **22**, 176001 (2010).

[48] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv:1412.6980.

[49] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, PyTorch: An imperative style, high-performance deep learning library, Adv. Neural Inf. Process. Syst. **32**, 8024 (2019).

[50] R. Ozawa, S. Hayami, K. Barros, G.-W. Chern, Y. Motome, and C. D. Batista, Vortex crystals with chiral stripes in itinerant magnets, J. Phys. Soc. Jpn. **85**, 103703 (2016).

[51] S. Hayami and Y. Motome, Topological spin crystals by itinerant frustration, J. Phys.: Condens. Matter **33**, 443001 (2021).

[52] 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).

[53] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, Geometric deep learning: Going beyond euclidean data, IEEE Signal Proc. Mag. **34**, 18 (2017).

[54] S. Chen, E. Dobriban, and J. H. Lee, A group-theoretic framework for data augmentation, J. Mach. Learn. Res. **21**, 9885 (2020).

[55] P. C. Nguyen, Y.-T. Nguyen, J. B. Choi, P. K. Seshadri, H. Udaykumar, and S. Baek, PARC: Physics-aware recurrent convolutional neural networks to assimilate meso scale reactive mechanics of energetic materials, Sci. Adv. **9**, eadd6868 (2023).

[56] P. C. Nguyen, J. B. Choi, H. Udaykumar, and S. Baek, Challenges and opportunities for machine learning in multiscale computational modeling, ASME J. Comput. Inf. Sci. Eng. **23**, 060808 (2023).

[57] P. C. Nguyen, Y.-T. Nguyen, P. K. Seshadri, J. B. Choi, H. Udaykumar, and S. Baek, A physics-aware deep learning model for energy localization in multiscale shock-to-detonation simulations of heterogeneous energetic materials, Propellants Explos. Pyrotech. **48**, e202200268 (2023).

[58] A. László, Unification mechanism for gauge and spacetime symmetries, J. Phys. A: Math. Theor. **50**, 115401 (2017).

[59] T. Dao, A. Gu, A. Ratner, V. Smith, C. De Sa, and C. Ré, A kernel theory of modern data augmentation, in *International Conference on Machine Learning* (JMLR, Cambridge, MA, 2019), pp. 1528–1537.

[60] R. Wang, R. Walters, and R. Yu, Data augmentation vs. equivariant networks: A theory of generalization on dynamics forecasting, arXiv:2206.09450.

[61] W. Luo, Y. Li, R. Urtasun, and R. Zemel, Understanding the effective receptive field in deep convolutional neural networks, Adv. Neural Inf. Process. Syst. **29** (2016).

[62] M. Geiger and T. Smidt, e3nn: Euclidean neural networks, arXiv:2207.09453.