

Scalable learning of macroscopic stochastic dynamics

Mengyi Chen ¹, Pengru Huang,² Kostya S. Novoselov ^{2,3} and Qianxiao Li ^{1,2,*}

¹Department of Mathematics, *National University of Singapore*, Singapore, Singapore

²Institute for Functional Intelligent Materials, *National University of Singapore*, Singapore, Singapore

³Materials Science and Engineering, *National University of Singapore*, Singapore, Singapore



(Received 29 September 2025; accepted 26 February 2026; published 20 March 2026)

Macroscopic dynamical descriptions of complex physical systems are crucial for understanding and controlling material behavior. With the growing availability of data and compute, machine learning has become a promising alternative to first-principles methods to build accurate macroscopic models from microscopic trajectory simulations. However, for spatially extended systems, direct simulations of sufficiently large microscopic systems that inform macroscopic behavior are prohibitive. In this work, we propose a framework that learns the macroscopic dynamics of large stochastic microscopic systems using only small-system simulations. Our framework employs a partial evolution scheme to generate training data pairs by evolving large-system snapshots within local patches. We subsequently derive the closure variables associated with the macroscopic observables and learn the macroscopic dynamics using a custom loss. Furthermore, we introduce a hierarchical upsampling scheme that enables the efficient generation of large-system snapshots from small-system snapshots. We empirically demonstrate the accuracy and robustness of our framework through a variety of stochastic spatially extended systems, including those described by stochastic partial differential equations, idealized lattice spin systems, and a more realistic NbMoTa alloy system.

DOI: [10.1103/mlh4-htxv](https://doi.org/10.1103/mlh4-htxv)

I. INTRODUCTION

Macroscopic observables characterize the collective behavior of complex microscopic dynamics and play a crucial role in real-world applications. They are typically functions of the full microscopic system. For example, magnetization is defined as the average of the local magnetic moments, and temperature as the average kinetic energy per degree of freedom. For alloy systems, macroscopic observables such as thermal conductivity, electrical conductivity, and magnetization are governed by a variety of microscopic interactions, including collective electron scattering mechanisms, lattice vibrations, and microscopic spin coupling interactions [1]. These macroscopic observables capture the material's overall behavior and functionality.

To obtain accurate time evolution of macroscopic observables, large-scale microscopic simulations over extended times are often required, yet their high computational expense remains the main bottleneck [2–4]. For example, density functional theory (DFT) and the related *ab initio* molecular dynamics (AIMD) represent a milestone in computational methods for studying molecules and solid-state materials at the quantum mechanical level. However, because of the well-known exponential wall challenge, where computational

expense increases rapidly with the number of particles, DFT simulations are usually limited to relatively small systems that may be insufficient for capturing true macroscopic behavior [5,6].

Various approaches have been developed to overcome the computational bottleneck. The kinetic Monte Carlo (KMC) algorithm represents the microscopic dynamics as a Markov chain by coarse-graining the time axis [7–9]. The transition rates for all possible events need to be computed for every KMC step. Therefore, KMC is still computationally prohibitive for large systems [10,11]. Machine learning force fields (MLFFs) replace the expensive *ab initio* force computations with efficient neural network-based predictions [12–15]. When applying MLFFs to molecular dynamics simulations, the time step of the molecular dynamics simulation must be chosen on the scale of femtoseconds to capture atomic vibrations. Consequently, millions to billions of integration steps are required for the simulation, and atomic forces are computed for each time step, still resulting in high computational cost for large systems [16]. Coarse-grained MLFFs further improve computational efficiency by mapping several atoms onto effective particles, thereby reducing the number of degrees of freedom. However, the training of coarse-grained MLFFs requires microscopic simulation data, and the largest barrier to applying coarse-grained MLFFs to large systems is generating enough training data [17–21]. The closure modeling methods, instead, model the dynamics of macroscopic observables directly, but still rely on short microscopic simulations of the large system or microscopic forces on all atoms for macroscopic dynamics derivation of large systems [22].

Despite their methodological differences, existing approaches all rely either on direct microscopic simulations or

*Contact author: qianxiao@nus.edu.sg

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

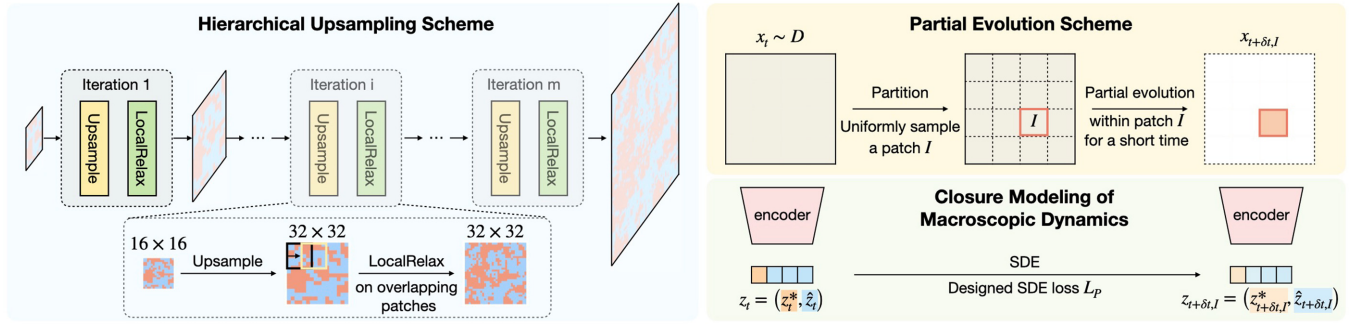


FIG. 1. Schematic illustration of our framework. The hierarchical upsampling scheme generates the large-system dataset D from the small-system dataset D_s through multiple iterations, each consisting of an UPSAMPLE and a LOCALRELAX step. An example of one iteration for the Ising model is shown. For the partial evolution scheme, for every $\mathbf{x}_t \in D$, a patch $\mathcal{I}$ is first uniformly sampled, then the microscopic dynamics is evolved locally within the patch $\mathcal{I}$ for a short time to yield $\mathbf{x}_{t+\delta t, \mathcal{I}}$. For the closure modeling, an autoencoder is trained to discover the closure variables to the macroscopic observables, and the macroscopic dynamics are derived with the designed loss $\mathcal{L}_p$.

on training data derived from such simulations. However, because of the computational constraint, microscopic simulation of large systems with millions to billions of atoms over extended time is generally intractable. This leads to a natural question: Can accurate macroscopic dynamics of large systems be obtained when only small-system microscopic simulations are accessible? Chen and Li [23] proposed a training procedure on the microscopic coordinates that addresses this question for deterministic dynamics. However, their method requires partial computation of microscopic forces for the macroscopic dynamics derivation. In the case of stochastic systems, where the dynamics are described by the conditional distribution of the next configuration given the current one, such microscopic forces are generally not well defined. Hence, their method cannot be easily generalized to stochastic systems. Yet, stochastic microscopic systems are arguably more prevalent, especially in the modeling of chemical reactions, molecular dynamics, and ferromagnetic phase transitions [24]. The main goal of our method is to address the above question for stochastic dynamical systems.

In this work, we develop a framework that can accurately derive the macroscopic dynamics of stochastic microscopic systems, while requiring only microscopic simulations of small systems. Specifically, given the dataset D , composed of snapshots of large systems spanning states from far-from-equilibrium to near-equilibrium, we introduce a partial evolution scheme that evolves $\mathbf{x}_t \in D$ locally within a local patch $\mathcal{I}$ for a short time δt to produce training data pairs $\{\mathbf{x}_t, \mathbf{x}_{t+\delta t, \mathcal{I}}\}$. Next, building on the workflow of Ref. [22], we derive the closure variables to the macroscopic observables and model the resulting dynamics via stochastic differential equations (SDE). To account for the additional stochasticity introduced by the random selection of patches in the partial evolution scheme, we introduce a modified SDE loss and provide a theoretical justification. In addition, we design a hierarchical upsampling scheme that efficiently generates a large-system dataset D from a small-system dataset D_s , which consists of snapshots sampled from trajectories of the small system.

The key idea of our framework is illustrated in Fig. 1. Key notations are summarized in Appendix A. We provide a detailed description of each component in Sec. II. In Sec. III,

we further validate the accuracy and robustness of our method through a variety of stochastic microscopic systems, including stochastic partial differential equations (SPDE) systems, spin systems, and a more realistic NbMoTa alloy system.

II. METHODOLOGY

We focus on microscopic systems that are spatially extended, including SPDE systems, the Ising model, and alloy systems. We assume the microscopic time evolution can be modeled as a Markov process of a random variable supported on a finite but large lattice structure. Let the microscopic state be $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_n) \in \mathbb{R}^n$, where n represents the number of lattice sites of the system. We assume the lattice sites are arranged on a regular lattice structure, and $\mathbf{x}_i$ represents some physical quantity associated with the i th lattice site. To better illustrate this, we provide several examples. In SPDE systems, $\mathbf{x}$ can be the state variables after spatial discretization on a regular grid, with n denoting the total number of grid points. In the Ising model, n denotes the number of spins, and $\mathbf{x}_i \in \{-1, 1\}$ represents the spin state of the i th spin. The spins are arranged on a square lattice. In alloy systems, n denotes the number of atoms and $\mathbf{x}_i$ represents the atom type of the i th lattice site. The atoms are arranged on a regular lattice structure, depending on the crystal structure of the alloy, such as body-centered cubic (BCC), face-centered cubic (FCC), hexagonal close-packed (HCP), or diamond cubic.

In many real applications, we are interested in the dynamics of some macroscopic observables, denoted by $\mathbf{z}^* = \boldsymbol{\varphi}^*(\mathbf{x})$. The form of $\boldsymbol{\varphi}^*$ is given beforehand, and $\boldsymbol{\varphi}^*$ can be applied to different system sizes. Since the underlying microscopic system is spatially extended, we are interested in the intensive quantities that do not scale with system size. For instance, in the Ising model, it is common to study the average magnetization $M = \sum_{i=1}^n \mathbf{x}_i / n$ instead of the total magnetization. In what follows, we will limit our discussion of macroscopic observables to intensive quantities.

Assume we are given a microscopic simulator $\mathcal{S}_{n_s}$, which can accurately simulate the microscopic dynamics of a small system up to size $n_s \ll n$ owing to computational constraints. From this simulator, we obtain the dataset D_s of the small system composed of snapshots sampled from trajectories of

the small system. The goal of this work is to derive the macroscopic dynamics of a large system of size n using only such small-scale simulations.

A. Closure modeling of macroscopic dynamics

Existing works on macroscopic dynamics derivation typically involve two components: discovering the closures for macroscopic observables, and jointly deriving their dynamics [22,25,26]. Our work adopts the same two components for macroscopic dynamics derivation. Assume we are given the dataset D of the large system, consisting of multiple snapshots of the large system, we will first generate temporal training data pairs as follows.

Partial evolution scheme. We propose a scheme for locally evolving the microscopic dynamics of a large system within a small spatial patch, which we refer to as the *partial evolution scheme*. The purpose of this scheme is to generate locally evolved training data pairs $\{\mathbf{x}_t, \mathbf{x}_{t+\delta t, \mathcal{I}}\}$ by evolving the microscopic dynamics on a small patch for a short time interval.

More specifically, we partition the underlying regular lattice into $K = n/n_s$ small patches, each containing n_s lattice sites. For example, for the two-dimensional Ising model on a 64^2 square lattice, we partition the large square into 64 square patches of size $n_s = 8^2$. Let the index set of lattice sites in the k th patch be $\mathcal{I}^k$, which is a subset of $\{1, \dots, n\}$ and contains n_s lattice sites. The state of the lattice sites in patch $\mathcal{I}^k$ is then written as $\mathbf{x}_{\mathcal{I}^k} = \{\mathbf{x}_i\}_{i \in \mathcal{I}^k}$, and the microscopic state as $\mathbf{x} = (\mathbf{x}_{\mathcal{I}^1}, \dots, \mathbf{x}_{\mathcal{I}^K})$. For each configuration $\mathbf{x}_t \sim D$, we uniformly sample a patch $\mathcal{I}$ from the K patches with probability $1/K$. Next, the microscopic simulator $\mathcal{S}_{n_s}$ is used to evolve $\mathbf{x}_t$ within the selected patch $\mathcal{I}$ for a short time δt , yielding the updated state $\mathbf{x}_{t+\delta t, \mathcal{I}}$. Consequently, the resulting training data pair is $\{\mathbf{x}_t, \mathbf{x}_{t+\delta t, \mathcal{I}}\}$. We denote the resulting conditional distribution of $\mathbf{x}_{t+\delta t, \mathcal{I}}$ as $q(\mathbf{x}_{t+\delta t, \mathcal{I}}|\mathbf{x}_t)$. Note that the time step δt may also be random. For instance, δt is sampled from an exponential distribution in the case of kinetic Monte Carlo dynamics.

The partial evolution scheme is related to patch dynamics in the equation-free framework (EFF) for multiscale simulation [27–29], in that both approaches evolve the microscopic dynamics on small spatial patches for short time intervals. Despite this similarity, we adopt it for a different purpose. The EFF makes use of patch dynamics as a simulation tool to advance macroscopic observables through repeated microscopic simulations, without explicitly deriving macroscopic dynamical equations. In contrast, we use partial evolution scheme as a data-generation mechanism for learning macroscopic dynamics.

We note that for the partial evolution scheme to be accurate, the microscopic dynamics within a small patch must evolve as if embedded in the full system. To achieve this, when evolving the microscopic dynamics within a local patch, appropriate boundary conditions should be imposed. In our implementation, the local environment is handled in two possible ways. One option is to treat the neighboring sites outside the patch as ghost cells, whose values are held fixed during the short-time evolution. This approach does not increase the computational cost of the partial evolution scheme. Alternatively, one may introduce a thin buffer region surrounding

the patch. In this approach, the patch and buffer evolve simultaneously, but only the central states are retained for the macroscopic derivation. The purpose of the buffer is to shield the interior dynamics from artificial boundary effects. This strategy is commonly used in the patch dynamics [30–32]. While this strategy slightly increases the computational cost, we will keep the buffer thin relative to the full system size, ensuring the cost of the partial evolution scheme remains low.

We will demonstrate how to derive the macroscopic dynamics from the training data pairs obtained from the partial evolution scheme.

Autoencoder for discovering closure variables. We employ an autoencoder architecture to discover closure variables $\hat{\mathbf{z}}$ associated with the macroscopic observables $\mathbf{z}^*$. The closure variables will capture the unresolved information by $\mathbf{z}^*$, and ensure the dynamics of $\mathbf{z} = (\mathbf{z}^*, \hat{\mathbf{z}})$ depend only on itself. For example, in our experiments, the dynamics of the Curie-Weiss model can be fully described by the magnetization, hence no closure variables will be needed if $\mathbf{z}^*$ represents the magnetization. In contrast, for the Ising model, the magnetization alone cannot capture all the dynamical information, and additional closure variables are required so that the dynamics of $\mathbf{z}$ only depend on itself.

Since the training data pairs take the form of $\{\mathbf{x}_t, \mathbf{x}_{t+\delta t, \mathcal{I}}\}$, we want the closure function to be well defined for both the microscopic state $\mathbf{x}$ and the microscopic state $\mathbf{x}_{\mathcal{I}}$, which are of different dimensions. Denote the closure function by $\hat{\phi}$. To achieve this, $\hat{\phi}$ is directly applied to $\mathbf{x}_{\mathcal{I}}$, yielding $\hat{\phi}(\mathbf{x}_{\mathcal{I}})$. The closure representation for the full state $\mathbf{x}$ is defined as the average of $\hat{\phi}(\mathbf{x}_{\mathcal{I}})$ over all the patches,

$$\hat{\phi}(\mathbf{x}) = \frac{1}{K} \sum_{\mathcal{I} \in \{\mathcal{I}^1, \dots, \mathcal{I}^K\}} \hat{\phi}(\mathbf{x}_{\mathcal{I}}). \quad (1)$$

We denote the closure variables by $\hat{\mathbf{z}} = \hat{\phi}(\mathbf{x})$, and concatenate it with the macroscopic observable $\mathbf{z}^* = \phi^*(\mathbf{x})$ to form the full latent state $\mathbf{z} = (\mathbf{z}^*, \hat{\mathbf{z}}) = (\phi^*(\mathbf{x}), \hat{\phi}(\mathbf{x}))$. By definition of the closure variables, $\hat{\mathbf{z}}$ are intensive quantities, just like the macroscopic observables $\mathbf{z}^*$. Therefore, $\mathbf{z}$ represents intensive quantities. Denote the encoder by $\phi = (\phi^*, \hat{\phi})$ and the decoder by ψ , where ϕ^* is the predefined macroscopic observable function with no trainable parameters. The functions $\hat{\phi}$ and ψ are parameterized by neural networks and are trained jointly. We omit the explicit dependence on the parameters for notational simplicity. The autoencoder is trained by minimizing the reconstruction loss,

$$\mathcal{L}_{\text{recon}} = \mathbb{E}_{\mathbf{x}_t} \|\psi \circ \phi(\mathbf{x}_t) - \mathbf{x}_t\|_2^2. \quad (2)$$

Once the autoencoder is trained, we will generate the latent training data pairs $\{\mathbf{z}_t, \mathbf{z}_{t+\delta t, \mathcal{I}}\}$ for the macroscopic dynamics derivation,

$$\begin{aligned} \mathbf{z}_t &= \phi(\mathbf{x}_t), \\ \mathbf{z}_{t+\delta t, \mathcal{I}} &:= \mathbf{z}_t + (\phi(\mathbf{x}_{t+\delta t, \mathcal{I}}) - \phi(\mathbf{x}_t, \mathcal{I})), \end{aligned} \quad (3)$$

where $\mathbf{x}_{t, \mathcal{I}}$ denotes the restriction of $\mathbf{x}_t$ to the local patch $\mathcal{I}$. Next, we introduce the process of deriving macroscopic dynamics.

Macroscopic dynamics derivation. We model the macroscopic dynamics with SDE,

$$d\mathbf{z}_t = \boldsymbol{\mu}(\mathbf{z}_t)dt + \boldsymbol{\Sigma}^{1/2}(\mathbf{z}_t)d\mathbf{B}_t, \quad (4)$$

where $\boldsymbol{\mu}$ is the drift term and $\boldsymbol{\Sigma}$ is the diffusion term. In most experiments, we adopt fully connected networks for both $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. For the NbMoTa alloy experiment in Sec. III C, we adopt OnsagerNet [22,33] for $\boldsymbol{\mu}$ to account for more complex macroscopic dynamics. The OnsagerNet can capture physically interpretable and stable macroscopic dynamics by incorporating the generalized Onsager principle into the model structure, and its form is given by

$$\boldsymbol{\mu}(\mathbf{z}_t) = -(\mathbf{M}(\mathbf{z}_t) + \mathbf{W}(\mathbf{z}_t))\nabla V(\mathbf{z}_t) + \mathbf{f}(\mathbf{z}_t), \quad (5)$$

where $\mathbf{M}$ is a symmetric positive semi-definite matrix describing energy dissipation, $\mathbf{W}$ is a skew-symmetric matrix describing energy conservation, V is a potential function, and $\mathbf{f}$ is a vector field representing external forces.

Existing works train the SDE by minimizing the negative log-likelihood [22,34–36],

$$\mathcal{L}[\boldsymbol{\mu}, \boldsymbol{\Sigma}] = \mathbb{E}_{\mathbf{z}_t, \mathbf{z}_{t+\delta t}} [-2 \log p(\mathbf{z}_{t+\delta t} | \mathbf{z}_t + \boldsymbol{\mu}(\mathbf{z}_t)\delta t, \boldsymbol{\Sigma}(\mathbf{z}_t)\delta t)], \quad (6)$$

where $\mathbf{z}_{t+\delta t}$ denotes the latent state obtained by evolving the full system from $\mathbf{z}_t$ over a time step δt . The conditional distribution p is given by the Gaussian distribution $\mathcal{N}(\mathbf{z}_{t+\delta t}; \mathbf{z}_t + \boldsymbol{\mu}(\mathbf{z}_t)\delta t, \boldsymbol{\Sigma}(\mathbf{z}_t)\delta t)$, obtained by discretizing the SDE with the Euler-Maruyama scheme.

In our setting, however, $\mathbf{z}_{t+\delta t, \mathcal{I}}$ is not obtained by evolving the full system by δt . Instead, it results from a partial evolution over a localized spatial patch. To account for this, we adapt the SDE loss as follows:

$$\mathcal{L}_p[\boldsymbol{\mu}, \boldsymbol{\Sigma}] = \mathbb{E}_{\mathbf{z}_t, \mathbf{z}_{t+\delta t, \mathcal{I}}} [-2 \log p(\mathbf{z}_{t+\delta t, \mathcal{I}} | \mathbf{z}_t + \boldsymbol{\mu}(\mathbf{z}_t)\delta t, K\boldsymbol{\Sigma}(\mathbf{z}_t)\delta t)]. \quad (7)$$

The only difference between $\mathcal{L}$ and $\mathcal{L}_p$ is that the covariance term in $\mathcal{L}_p$ is multiplied by a factor K , where K denotes the number of patches introduced earlier. We can interpret the influence of K qualitatively. The factor K in the loss compensates for the additional stochasticity, leading to a smaller learned diffusion term. During the data generation of $\mathbf{x}_{t+\delta t, \mathcal{I}}$ from $\mathbf{x}_t$, we introduce additional randomness by performing partial evolution. Therefore, in the derivation of macroscopic dynamics, we multiply the diffusion term by K to correct for the extra stochasticity.

To learn the macroscopic dynamics, we parametrize both $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ with neural networks, denoted by $\boldsymbol{\mu}_\theta$ and $\boldsymbol{\Sigma}_\theta$, respectively. The training objective is given by the loss function,

$$\mathcal{L}_p(\theta) := \mathcal{L}_p[\boldsymbol{\mu}_\theta, \boldsymbol{\Sigma}_\theta]. \quad (8)$$

We minimize $\mathcal{L}_p(\theta)$ to obtain the optimal parameter.

It is important to observe that in Eq. (3), we define $\mathbf{z}_{t+\delta t, \mathcal{I}}$ in a nontrivial way, while the most naive way will be $\mathbf{z}_{t+\delta t, \mathcal{I}} = \boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{I}})$. Note that the loss function $\mathcal{L}_p(\theta)$ involves the weighted norm of the residual $\mathbf{z}_{t+\delta t, \mathcal{I}} - \mathbf{z}_t - \boldsymbol{\mu}(\mathbf{z}_t)\delta t$ since p represents a Gaussian distribution. If we define $\mathbf{z}_{t+\delta t, \mathcal{I}} =$

$\boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{I}})$ in the naive way, we will have

$$\mathbf{z}_{t+\delta t, \mathcal{I}} - \mathbf{z}_t = \frac{1}{K} \sum_{\mathcal{J}} (\boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{I}}) - \boldsymbol{\varphi}(\mathbf{x}_{t, \mathcal{J}})), \quad (9)$$

which is the average of $\boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{I}}) - \boldsymbol{\varphi}(\mathbf{x}_{t, \mathcal{J}})$. Since $\boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{I}})$ may be very different from $\boldsymbol{\varphi}(\mathbf{x}_{t, \mathcal{J}})$ when $\mathcal{J} \neq \mathcal{I}$, the resulting $\mathbf{z}_{t+\delta t, \mathcal{I}} - \mathbf{z}_t$ will be very noisy. However, the definition in Eq. (3) will yield

$$\mathbf{z}_{t+\delta t, \mathcal{I}} - \mathbf{z}_t = \boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{I}}) - \boldsymbol{\varphi}(\mathbf{x}_{t, \mathcal{I}}), \quad (10)$$

which directly measures the change in the patch $\mathcal{I}$, thus reducing noise and leading to more stable training. We refer to the approach that adopts the naive formulation $\mathbf{z}_{t+\delta t, \mathcal{I}} = \boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{I}})$ and derives the macroscopic dynamics via $\mathcal{L}$ as the baseline. A detailed comparison of our method with the baseline is provided in Sec. III.

Theoretical justification. In many microscopic systems, the microscopic interactions between lattice sites are local. For instance, microscopic interactions are limited to the first-nearest neighbor in the Ising model, and are limited to a finite cutoff distance in alloy systems. Under this locality assumption, when the time increment δt is sufficiently small, the state increments on disjoint spatial patches are approximately independent. Specifically, define the short-time increments

$$\begin{aligned} \Delta \mathbf{z}_t &:= \mathbf{z}_{t+\delta t} - \mathbf{z}_t, & \Delta \mathbf{z}_t^* &:= \mathbf{z}_{t+\delta t}^* - \mathbf{z}_t^*, \\ \Delta \mathbf{z}_{t, \mathcal{I}} &:= \mathbf{z}_{t+\delta t, \mathcal{I}} - \mathbf{z}_t, & \Delta \mathbf{z}_{t, \mathcal{I}}^* &:= \mathbf{z}_{t+\delta t, \mathcal{I}}^* - \mathbf{z}_t^*. \end{aligned} \quad (11)$$

Then, for two disjoint patches $\mathcal{I} \neq \mathcal{J}$, we have

$$\begin{aligned} \Delta \mathbf{z}_{t, \mathcal{I}} &= \boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{I}}) - \boldsymbol{\varphi}(\mathbf{x}_{t, \mathcal{I}}), \\ \Delta \mathbf{z}_{t, \mathcal{J}} &= \boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{J}}) - \boldsymbol{\varphi}(\mathbf{x}_{t, \mathcal{J}}), \end{aligned} \quad (12)$$

which can be treated as approximately independent random variables.

Let $\hat{q}(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t)$ denote the distribution of $\mathbf{x}_{t+\delta t, \mathcal{I}}$ derived by evolving the full system from $\mathbf{x}_t$ for a time step δt and subsequently restricting the state to the local patch $\mathcal{I}$. Under the local interaction assumption and for sufficiently small δt , the distribution q can be well approximated by the partial evolution distribution $\hat{q}$,

$$q(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t) \approx \hat{q}(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t). \quad (13)$$

Furthermore, since we consider a sufficiently large microscopic system and the macroscopic observables are intensive quantities, the macroscopic observable of the full system can be approximated by the average over local patches,

$$\boldsymbol{\varphi}^*(\mathbf{x}) \approx \frac{1}{K} \sum_{\mathcal{I}} \boldsymbol{\varphi}^*(\mathbf{x}_{\mathcal{I}}). \quad (14)$$

Similarly, the macroscopic increment $\Delta \mathbf{z}_t^* = \boldsymbol{\varphi}^*(\mathbf{x}_{t+\delta t}) - \boldsymbol{\varphi}^*(\mathbf{x}_t)$ can be approximated by the average of the local increments,

$$\Delta \mathbf{z}_t^* \approx \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t, \mathcal{I}}^*. \quad (15)$$

Equations (14) and (15) hold exactly when the macroscopic observable can be written as the mean of a function depending only on individual lattice sites, such as the magnetization. For more complex macroscopic observables, the equality is only

approximate, but the approximation improves and becomes accurate in the limit of a large system size.

We provide a theoretical justification for the loss $\mathcal{L}_p$ under the above conditions:

Theorem 1. Assume that for any two disjoint patches $\mathcal{I} \neq \mathcal{J}$, the short-time increments $\Delta \mathbf{z}_{t,\mathcal{I}}$ and $\Delta \mathbf{z}_{t,\mathcal{J}}$ are conditionally independent given $\mathbf{x}_t$. Assume further that the encoder $\boldsymbol{\varphi}$ is uniformly bounded, i.e., $\|\boldsymbol{\varphi}\|_\infty \leq M$ for some $M > 0$.

Suppose that Eqs. (13) and (15) hold up to second-order accuracy in δt , in the sense that the total variation distance between q and $\hat{q}$ satisfies

$$\delta_{\text{TV}}(q(\cdot | \mathbf{x}_t), \hat{q}(\cdot | \mathbf{x}_t)) \leq C_1 \delta t^2, \quad (16)$$

and that the macroscopic increment approximation error satisfies

$$\left\| \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} [\Delta \mathbf{z}_t^* - \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t,\mathcal{I}}^*] \right\| \leq C_2 \delta t^2, \quad (17)$$

where $C_1, C_2 > 0$ are constants and $\mathbf{x}_{t+\delta t} | \mathbf{x}_t$ denotes the ground-truth conditional distribution obtained by evolving the full system from $\mathbf{x}_t$ for time δt .

Under these assumptions, the functional $\mathcal{L}[\boldsymbol{\mu}, \boldsymbol{\Sigma}]$ admits a unique minimizer $(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$, and the functional $\mathcal{L}_p[\boldsymbol{\mu}, \boldsymbol{\Sigma}]$ admits a unique minimizer $(\boldsymbol{\mu}^\dagger, \boldsymbol{\Sigma}^\dagger)$, such that

$$\begin{aligned} \|\boldsymbol{\mu}^* - \boldsymbol{\mu}^\dagger\|_\infty &\leq K_1 \delta t, \\ \|\boldsymbol{\Sigma}^* - \boldsymbol{\Sigma}^\dagger\|_\infty &\leq K_2 \delta t, \end{aligned} \quad (18)$$

where $K_1, K_2 > 0$ are constants depending only on C_1, C_2 , and M . In particular, if $q = \hat{q}$ holds exactly, we have

$$\boldsymbol{\mu}^* = \boldsymbol{\mu}^\dagger, \quad \boldsymbol{\Sigma}^* = \boldsymbol{\Sigma}^\dagger. \quad (19)$$

We emphasize that the independence assumption in Theorem 1 is about short-time increments on disjoint patches conditioned on $\mathbf{x}_t$, not about independence of the patch states themselves. The proof is based on a direct computation of the first and second variations of the loss functions. We provide the full proof in Appendix C. Theorem 1 theoretically justifies our framework: Under appropriate conditions, the macroscopic dynamics learned from data generated via partial evolution are as accurate as those learned from full, computationally expensive microscopic simulations. In particular, for the stochastic predator-prey system in Sec. III A, the assumptions of Theorem 1 are satisfied, as verified in Appendix E.

B. Hierarchical upsampling scheme

In Sec. II A, we assume access to the dataset D , which contains multiple snapshots of large systems. In practice, however, direct access to D is typically unavailable, as only microscopic simulations of small systems can be performed. To address this, we introduce a hierarchical upsampling scheme for generating the large-system dataset D from the small-system dataset D_s . The hierarchical upsampling scheme is illustrated in Algorithm 1. It consists of multiple iterations, each involving two steps: UPSAMPLE and LOCALRELAX. In the UPSAMPLE step, the configurations in $D^{(i)}$ are expanded into configurations of size m times larger, where $m \geq 2$ is an integer. Next, we apply a LOCALRELAX step to remove the unphysical artifacts that are introduced in the UPSAMPLE step.

ALGORITHM 1. Hierarchical upsampling scheme.

Require:

D_s : dataset of small-system snapshots
 N_{iter} : number of iterations
1: Initialize $D^{(0)} \leftarrow D_s$
2: **for** $i = 1$ **to** N_{iter} **do**
3: $D^{(i)} \leftarrow \text{LOCALRELAX}(\text{UPSAMPLE}(D^{(i-1)}))$
4: $D \leftarrow D^{(N_{\text{iter}})}$
5: **return** D $\triangleright$ dataset of large-system snapshots

More specifically, each generated large-system configuration is divided into overlapping patches of size n_s , and short-time relaxation or local dynamics evolution is applied within each patch.

For example, consider the two-dimensional Ising model to be introduced in Sec. III B, where the microscopic dynamics is chosen to be the continuous-time Glauber dynamics. We assume direct simulations are feasible only for systems up to size $n_s = 8^2$. Starting from the small-system dataset D_s , we apply the hierarchical upsampling scheme for $N_{\text{iter}} = 3$ iterations to obtain the large-system dataset D of size $n = 64^2$. In the first iteration, the UPSAMPLE step replicates each spin into a $m = 2^2$ block, yielding a 16^2 dimensional system. Next, we apply the LOCALRELAX step by dividing each 16^2 dimensional configuration into 16 patches of size $n_s = 8^2$ with a stride of 4. Within each patch, we run the continuous-time Glauber dynamics for a short time to remove the unphysical artifacts introduced in the UPSAMPLE step. In the second iteration, starting from the 16^2 dimensional dataset $D^{(1)}$, we repeat the UPSAMPLE and LOCALRELAX step to obtain $D^{(2)}$ for the 32^2 dimensional system. Subsequently, in the third iteration, we obtain the target large-system dataset D of size 64^2 from $D^{(2)}$. We provide a graphical illustration of one iteration for the Ising model in Fig. 1(a). The concrete form of UPSAMPLE and LOCALRELAX may vary for different microscopic systems, and further details are given in Sec. III.

By generating training data through a hierarchical upsampling and partial evolution scheme, our method circumvents the expensive, large-system microscopic simulations that are required by most of the existing methods. The main computational savings of our method come from the efficient generation of training data.

III. RESULTS

In this section, we empirically validate the accuracy and robustness of our method across various microscopic systems. We first demonstrate our method on a SPDE system and spin systems, and then validate it on a more realistic NbMoTa alloy system.

A. Stochastic predator-prey system

We first consider a one-dimensional SPDE system, mainly to validate the correctness of our method and investigate the impact of the coefficient K in the designed loss $\mathcal{L}_p$. The stochastic predator-prey system is given by Eq. (20), where u, v denote the dimensionless populations of the prey and

predator, and ξ_u, ξ_v are independent space-time white noise terms. In our experiment, we set the parameters $a = 3, b = 0.4, c = 0$, and $\sigma_u = \sigma_v = 0.02$,

$$\begin{aligned}\frac{\partial u}{\partial t} &= u(1 - u - v) + c \frac{\partial^2 u}{\partial x^2} + \sigma_u \xi_u(t, x), \\ \frac{\partial v}{\partial t} &= av(u - b) + \frac{\partial^2 v}{\partial x^2} + \sigma_v \xi_v(t, x),\end{aligned}\quad (20)$$

$$x \in \Omega = [0, 1], t \geq 0.$$

We impose Neumann boundary conditions,

$$\frac{\partial u}{\partial x}(t, 0) = \frac{\partial u}{\partial x}(t, 1) = 0, \quad \frac{\partial v}{\partial x}(t, 0) = \frac{\partial v}{\partial x}(t, 1) = 0. \quad (21)$$

The initial conditions are defined as

$$\begin{aligned}u(x, 0) &= c_1 + c_2 \cos(10\pi x), \\ v(x, 0) &= c_1 - c_2 \cos(10\pi x),\end{aligned}\quad (22)$$

where $c_1 \sim \mathcal{U}(0.05, 0.15)$, $c_2 \sim \mathcal{U}(0.45, 0.55)$.

For training data generation, we first generate the small-system dataset D_s by discretizing the spatial domain Ω into 100 uniform grids. We then solve Eq. (20) from $t = 0$ to $T = 30$ with time step $\delta t = 0.01$. The small system thus contains $n_s = 100$ lattice sites. The goal is to learn the macroscopic observables of a large system discretized on $n = 200$ uniform grid points. To construct the large-system dataset D , we perform an UPSAMPLE step by linearly interpolating the solution from the coarse grid with 100 points onto a finer grid with 200 points. Compared to snapshots sampled from large-system trajectories, snapshots generated by linear interpolation are smoother. However, we expect the resulting differences to be small. Therefore, for simplicity, we do not apply the LOCALRELAX step.

The $n = 200$ uniform grids are partitioned into $K = 5$ patches, each containing 40 grids. For each $\mathbf{x}_t \sim D$, we first uniformly sample a patch with $p = 1/5$, and subsequently evolve the system locally within the patch for one time step $\delta t = 0.01$ to obtain the updated state $\mathbf{x}_{t+\delta t, \mathcal{I}}$.

The macroscopic observable $\mathbf{z}^*$ of interest is chosen to be the mean of the populations of the prey and predator over the spatial grid points, which is two dimensional. We derive another two closure variables using an autoencoder, hence the dimension of $\mathbf{z}$ is 4. The only difference between $\mathcal{L}$ and $\mathcal{L}_p$ lies in the coefficient multiplying the diffusion term. We treat the coefficient as a hyperparameter λ and train the SDE with different values of λ . Theoretical analysis in Theorem 1 shows that the optimal value of λ is K under appropriate conditions. We explore various values of λ to empirically investigate the influence of λ .

Figure 2 shows the result on the nine test datasets with different combinations of parameters $(c_1, c_2) \in \{0.05, 0.10, 0.15\} \times \{0.45, 0.50, 0.55\}$. Each test dataset consists of 50 trajectories with the same initial condition. Once the SDE model in Eq. (4) is trained, we simulate it for a long time with the Euler-Maruyama method starting from the same initial condition as the test dataset. We employ maximum mean discrepancy (MMD) [37,38] to quantify the discrepancy between the predicted and ground-truth trajectories, which is widely used for comparing probability distributions. We use a mixture of radial basis function (RBF)

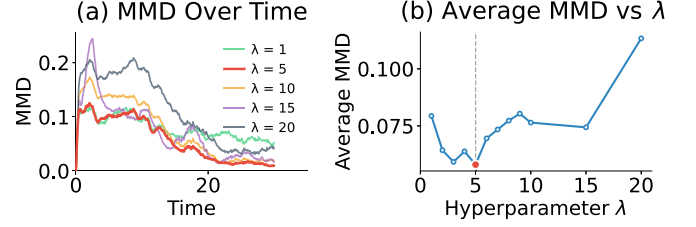


FIG. 2. Results on the stochastic predator-prey system. (a) The MMD is plotted as a function of time. (b) The average MMD over the entire simulation time is reported as a function of the hyperparameter λ .

kernels with varying bandwidths to improve the robustness of MMD. For each time point, we calculate the MMD between the marginal distributions of the predicted trajectories and the ground-truth trajectories, and report the results averaged over all nine test datasets.

From Fig. 2, we can observe that when $\lambda = K = 5$, the predicted trajectories achieve the minimal discrepancy from the ground-truth trajectories, which correspond well with our theoretical analysis. In fact, for the stochastic predator-prey system, the assumptions in Theorem 1 are exactly satisfied. Therefore, it is expected that the optimal hyperparameter λ is equal to K . In subsequent experiments with more complex microscopic dynamics, the assumptions in Theorem 1 may only hold approximately. Thus, the optimal value of λ may deviate slightly from K . We will treat λ as a tunable hyperparameter and perform a hyperparameter search initialized from K in the following experiments.

B. Ising model

Having validated our method on a toy SPDE system, we now turn to more complex spin systems. We first consider the Ising model, which plays an important role in statistical physics for investigating order-disorder phase transitions and critical phenomena [39,40]. The experiments are divided into two parts. In the first part, we conduct an ablation study on the computational power n_s of the microscopic simulator $\mathcal{S}_{n_s}$, and compare the performance of our method with the baseline across different values of n_s . In the second part, we further evaluate the ability of our method to accurately capture the critical behavior of macroscopic dynamics.

We consider the two-dimensional Ising model, where the spins are arranged on a square lattice of size $n = L \times L$. The Hamiltonian of the Ising model is given by

$$H_{n,h}(\sigma) = -\frac{J}{2} \sum_{\langle i,j \rangle} \sigma_i \sigma_j - h \sum_{i=1}^n \sigma_i, \quad (23)$$

where $J > 0$ denotes the interaction strength, h denotes the external magnetic field, $\langle i, j \rangle$ represents nearest-neighbor pairs, and $\sigma_i \in \{-1, 1\}$ denotes the spin at site i . The microscopic state is denoted by $\mathbf{x} = \{\sigma_1, \dots, \sigma_n\}$. Throughout this paper, we employ dimensionless units by setting $J = 1$ and the Boltzmann constant $k_B = 1$. We describe the microscopic evolution of the system using continuous-time Glauber dynamics, as detailed in Appendix B. The critical temperature of the two-dimensional Ising model is $T_c \approx 2.269$. The Ising model

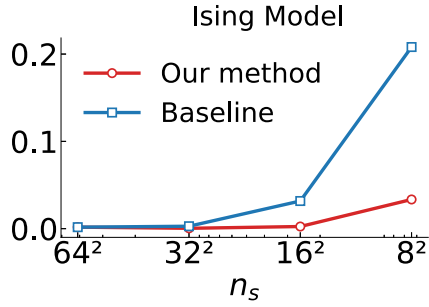


FIG. 3. Results on the Ising model, where the test error is plotted as a function of n_s . The test error is the mean relative error of the mean macroscopic observables between ground-truth and predicted trajectories.

exhibits an ordered ferromagnetic phase when $T < T_c$, and a disordered paramagnetic phase when $T > T_c$.

For the macroscopic observables, we consider the magnetization and domain wall density defined by

$$\rho_{\text{DW}} = \frac{1}{4L^2} \sum_{(i,j)} (1 - \sigma_i \sigma_j). \quad (24)$$

The domain wall density quantifies the degree of disorder by measuring the fraction of misaligned nearest-neighbor spin pairs. We derive two additional closure variables using an autoencoder, resulting in a latent state $\mathbf{z}$ of dimension 4.

In the first part, we fix the large-system size to $n = 64^2$, while varying the computational power of the microscopic simulator $\mathcal{S}_{n_s}$ by considering $n_s \in \{8^2, 16^2, 32^2, 64^2\}$. When $n_s = 64^2$, our method reduces to traditional methods for deriving the macroscopic dynamics. We set the external field to $h = 0.1$ and the temperature to $T = 2.5 > T_c$. To generate the large-system dataset D , we employ the hierarchical algorithm with $\log_2(n/n_s)$ iterations. For each iteration, we first perform an UPSAMPLE step to replicate every spin into 2^2 block. Next, we perform a LOCALRELAX step by evolving a short-time continuous-time Glauber dynamics locally to remove the unphysical artifacts introduced in the UPSAMPLE step.

Figures 3 and 4 compare the performance of our method with the baseline. From Fig. 3, we observe that our method consistently outperforms the baseline when $n_s < n$, which demonstrates the effectiveness of our method. Figure 4 further compares the magnetization statistics and domain-wall density statistics derived from ground-truth trajectories with those produced by the baseline model and our method. The statistics of the trajectories predicted by our method align closely with the ground truth, whereas the baseline exhibits much larger variations.

The parameter n_s can influence our method in two ways: (i) The size of the small-system dataset D_s is n_s . For smaller n_s , more iterations of upsampling are required to obtain D , which may degrade the data quality of D . (ii) When generating $\mathbf{x}_{t+\delta t, \mathcal{I}}$ from $\mathbf{x} \sim D$, we perform partial evolution on a small patch of size n_s . A smaller n_s leads to higher stochasticity of the generated data. As a result, a smaller n_s generally leads to worse performance. However, the test error of our method remains relatively low when $n_s \geq 16^2$, which demonstrates the robustness of our method with respect to n_s .

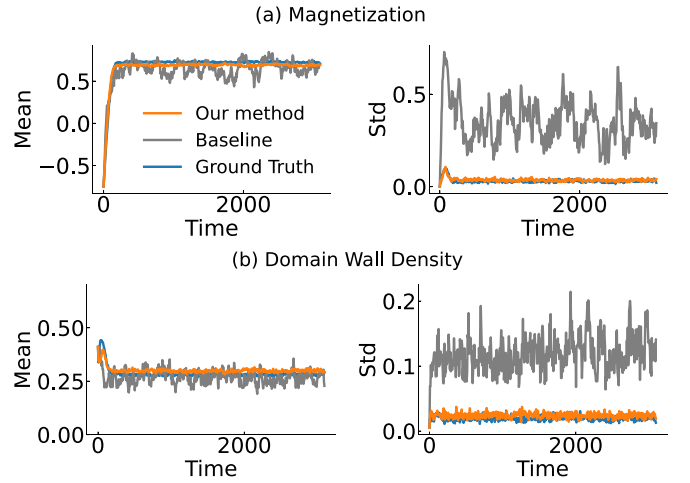


FIG. 4. Results on the Ising model with $n_s = 16^2$. Mean and standard deviation are estimated from 20 trajectories per method. (a) Magnetization statistics. (b) Domain wall density statistics.

In the second part of the experiment, we demonstrate that our method is capable of capturing the critical behavior of macroscopic dynamics and estimating the critical exponents. Finite-size scaling theory describes how equilibrium observables of a finite system with size $n = L \times L$ scale with L and T near the critical temperature. Specifically, for the equilibrium magnetization $\langle |M| \rangle$ and magnetic susceptibility $\mathcal{X}$, we will have [41]

$$\begin{aligned} \langle |M| \rangle(T, L) &= L^{-\beta/\nu} \mathcal{F}_M((T - T_c)L^{1/\nu}), \\ \mathcal{X}(T, L) &= L^{\gamma/\nu} \mathcal{F}_\mathcal{X}((T - T_c)L^{1/\nu}), \end{aligned} \quad (25)$$

where $\mathcal{F}_M$ and $\mathcal{F}_\mathcal{X}$ are scaling functions and the magnetic susceptibility $\mathcal{X}$ is defined as

$$\mathcal{X} = \frac{L^2}{T} (\langle M^2 \rangle - \langle |M| \rangle^2). \quad (26)$$

When $T = T_c$, Eq. (25) simplify to

$$\begin{aligned} \langle |M| \rangle(T_c, L) &\sim L^{-\beta/\nu}, \\ \mathcal{X}(T_c, L) &\sim L^{\gamma/\nu}. \end{aligned} \quad (27)$$

For the two-dimensional Ising model, the theoretical values of the critical exponents are $\beta = 1/8$, $\nu = 1$, $\gamma = 7/4$ [41].

In this part of the experiment, we fix the small-system size to be $n_s = 16^2$, while varying the large-system size $n = L \times L$. We apply our method across different system sizes and a range of temperatures near T_c . Next, we perform long-time simulations towards equilibrium using the trained SDE models. We calculate the equilibrium magnetization and magnetic susceptibility from the predicted trajectories, to which we fit the critical exponents.

In Figs. 5(a) and 5(b), we plot the equilibrium magnetization and susceptibility as a function of L at $T = 2.27$, which is very close to T_c . Next, we fit log-log curves to the data by Eq. (27), yielding estimated critical exponent $(\beta/\nu)^* = 0.129$, $(\gamma/\nu)^* = 1.670$. The estimated critical exponents are very close to the theoretical value $\beta/\nu = 0.125$, $\gamma/\nu = 1.75$. These results show that our method can accurately recover the ratios β/ν and γ/ν , demonstrating its ability to capture

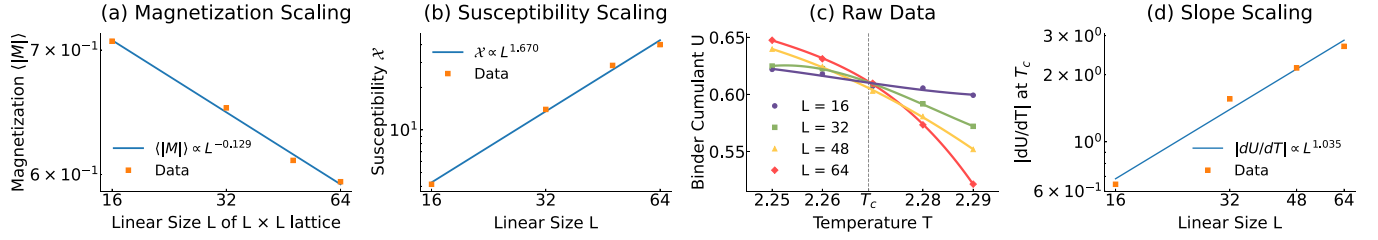


FIG. 5. Finite-size scaling results of the Ising model at $T = 2.27$, which is very close to $T_c \approx 2.269$. (a) Scaling of the equilibrium magnetization, with a fitted value $(\beta/\nu)^* = 0.129$. (b) Scaling of the magnetic susceptibility, with a fitted value $(\gamma/\nu)^* = 1.670$. (c) Raw Binder cumulant data for different temperatures and linear size L . Linear fitting is used for $L = 16$, and cubic polynomial fitting for $L = 32, 48, 64$. The slope at T_c is extracted from the fitted curve. (d) Log-log plot of $|dU/dT|_{T=T_c}$ versus L , with a fitted value $\nu^* = 1/1.035 \approx 0.97$.

accurate macroscopic dynamics across system sizes. This is especially significant because we only use microscopic simulations of small systems with size $n_s = 16^2$, yet our method still succeeds in reproducing macroscopic dynamics for large systems with different sizes.

We further evaluate whether our method can recover the individual critical exponents β , ν , γ . To achieve this, we estimate ν using the Binder cumulant, defined as

$$U(T, L) = 1 - \frac{\langle M^4 \rangle}{3\langle M^2 \rangle^2}. \quad (28)$$

According to finite-size scaling theory, the slope of the Binder cumulant at the critical temperature will scale with L as follows [41]:

$$|dU/dT|_{T=T_c} \sim L^{1/\nu}. \quad (29)$$

We plot the raw Binder cumulant data in Fig. 5(c) and compute the slope at T_c by fitting a curve to the data. In Fig. 5(d), a log-log fit gives the estimated $\nu^* = 0.97$, which is close to the theoretical value of $\nu = 1$.

From Eq. (25), we note that if we plot $\langle M \rangle L^{\beta/\nu}$ as a function of $(T - T_c) \cdot L^{1/\nu}$, the data should collapse onto the same curve for different T and L . Similarly, $\chi \cdot L^{-\gamma/\nu}$ and $(T - T_c) \cdot L^{1/\nu}$ should also collapse onto the same curve. We present these results in Fig. 6. We can observe that even though the raw magnetization data and raw susceptibility data lie on different curves for different T and L , the scaled data approximately collapse onto the same curve. This validates that our method can accurately capture the critical behavior of macroscopic dynamics in the Ising model.

C. NbMoTa alloy

We next validate our method on a more realistic NbMoTa equimolar alloy system to demonstrate its robustness in handling complex microscopic systems. We adopt the experimental setting of Ref. [42], where a neural network was trained to capture energy barriers and used to investigate microscopic diffusion dynamics.

The microscopic diffusion dynamics are modeled using the kinetic Monte Carlo (KMC) algorithm. In body-centered cubic (BCC) systems such as NbMoTa alloy, each vacancy i can jump to one of its eight first-nearest neighbors with rate $k_{ij} = k_0 \exp(-E_{ij}/k_B T)$, $j = 1, \dots, 8$, where E_{ij} is the energy barrier for the jump to the j th neighbor and k_0 is an attempt frequency. In our study, we employ the pretrained neural network model from Ref. [42] for energy barrier calculation. At each KMC step, the transition rates for all vacancies are computed, giving the total jump rate $R_{\text{tot}} = \sum_{i,j} k_{ij}$. Next, the time increment is sampled from the exponential distribution $\Delta t \sim -\frac{\ln r}{R_{\text{tot}}}$, $r \sim \mathcal{U}(0, 1)$. For large systems, the number of vacancies is substantial, and evaluating transition rates for all possible events is computationally demanding.

For the macroscopic observables, we investigate the nonproportional short-range order (SRO) parameters $\delta_{\text{Nb-Nb}}$, $\delta_{\text{Nb-Mo}}$, $\delta_{\text{Nb-Ta}}$, $\delta_{\text{Mo-Mo}}$, $\delta_{\text{Mo-Ta}}$, $\delta_{\text{Ta-Ta}}$. The SRO δ_{ij} is defined by

$$\delta_{i-j} = \frac{n_{ij} - n_{0,ij}}{n_i}, \quad (30)$$

where n_i is the number of atoms of type i , n_{ij} denotes the number of pairs between atom i and j in the first-nearest-neighbor shell, and $n_{0,ij}$ denotes the number of pairs in random solutions. Since we consider an equimolar NbMoTa alloy with

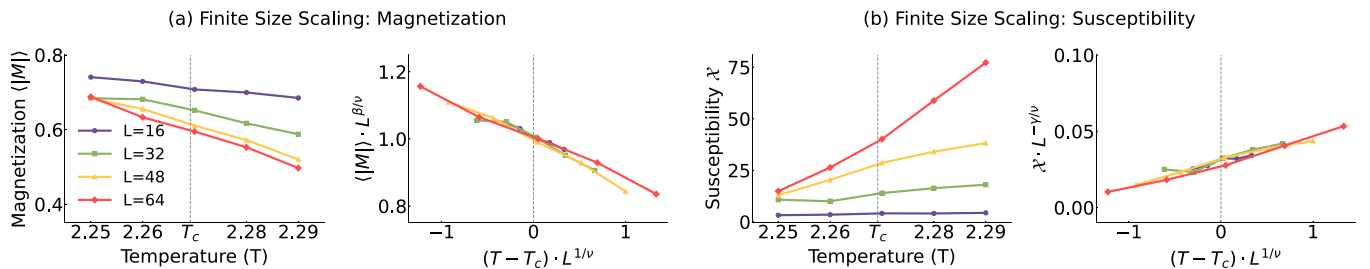


FIG. 6. Finite-size scaling results of the Ising model across various temperatures and system sizes. The data are collapsed using the theoretical critical exponents. (a) Finite-size scaling analysis of the magnetization data calculated from the predicted trajectories by our method. (b) Finite-size scaling analysis of the susceptibility.

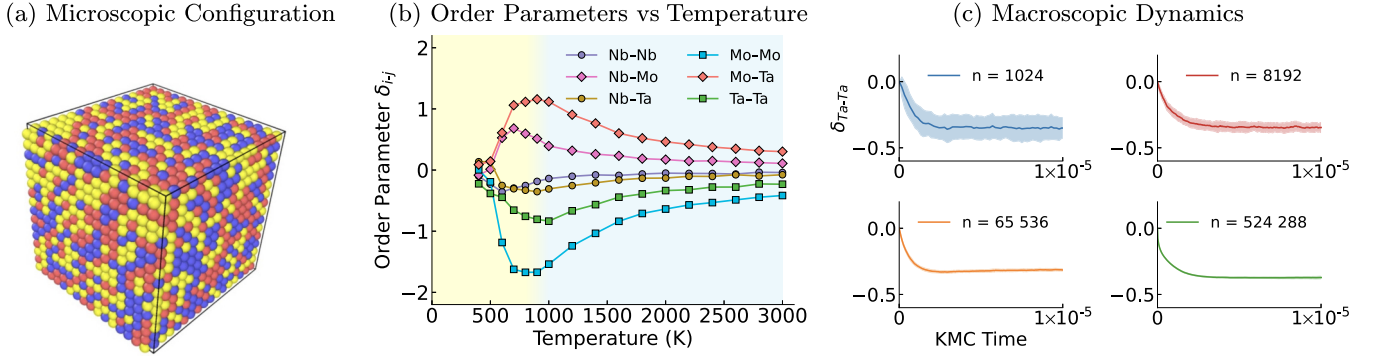


FIG. 7. Results of the NbMoTa equimolar alloy system. (a) Microscopic configuration of NbMoTa alloy with 8192 atoms. (b) Equilibrium order parameters as a function of temperature, obtained from macroscopic dynamics simulations by the trained SDE model. The microscopic system contains $n = 8192$ atoms. (c) Macroscopic dynamics of $\delta_{\text{Ta-Ta}}$ for microscopic systems of varying sizes when $T = 2000$ K. The mean and standard deviations are calculated over 100 trajectories.

BCC structure, $n_{0,ij} = 8/3$. For random configuration, $\delta_{i-j} = 0$. A positive δ_{i-j} indicates a favored $i-j$ pair, while a negative δ_{i-j} indicates an unfavored $i-j$ pair. The order parameter δ_{i-j} is essentially a rescaled form of the well-studied Warren-Cowley SRO parameter [43–45].

In our experiment, we set the small-system size to $n_s = 1024$, with a vacancy concentration corresponding to one vacancy per 1024 sites. Denote the supercell length by L , then the supercell will consist of $2L^3$ atoms for a BCC lattice. Hence, the supercell length of the small system is $L = 8$. Within the hierarchical upsampling scheme, an UPSAMPLE step is performed by concatenating multiple small-system configurations, followed by a LOCALRELAX step where KMC dynamics are run within small patches to remove unphysical artifacts. We first train an SDE model for the small system with $n_s = 1024$ atoms, and find that learning the macroscopic dynamics of the six macroscopic observables directly can already give accurate results consistent with microscopic simulations. Therefore, for the macroscopic dynamics derivation of the large system from data D , no closure variables are derived and the latent dimension is 6.

We first consider a large system with $n = 8192$ atoms. Training data are collected across temperatures from $T = 400$ K to $T = 3000$ K, and an SDE model dependent on T is trained,

$$d\mathbf{z}_t = \boldsymbol{\mu}(\mathbf{z}_t, T)dt + \boldsymbol{\Sigma}^{1/2}(\mathbf{z}_t, T)d\mathbf{B}_t. \quad (31)$$

The trained SDE is subsequently simulated for a long time at different temperatures. Figure 7(b) shows that the absolute value of $\delta_{\text{Mo-Ta}}$, $\delta_{\text{Ta-Ta}}$, $\delta_{\text{Mo-Mo}}$ increases for $T < 800$ K, reaches a maximum around $T = 800$ – 900 K, and decreases thereafter. The results indicate a critical temperature around $800 \sim 900$ K, corresponding to the regime of maximal diffusion-favored ordering. Our results correspond to the results obtained from the microscopic simulation in Ref. [42], confirming the accuracy and effectiveness of our method in detecting phase transitions.

To assess scalability, we further extend our method to much larger systems with $n = 65536$ ($L = 32$) and $n = 524288$ ($L = 64$) atoms at $T = 2000$ K. In Fig. 7(c), we show the macroscopic dynamics of $\delta_{\text{Ta-Ta}}$ for different system sizes at $T = 2000$ K. While larger systems require more KMC time

steps to reach equilibrium, each KMC step corresponds to a smaller time increment. From Fig. 7(c), we observe that the equilibrium KMC time is nearly identical across systems of different sizes. While the mean of the macroscopic dynamics converges approximately to the same value, the trajectories of smaller systems are more stochastic.

IV. DISCUSSION

This work proposes a framework to learn macroscopic dynamics of large microscopic systems from small-system simulations. We apply our method to SPDEs, spin systems, and an NbMoTa alloy system, scaling it to a large system with 524 288 atoms. Through these applications, we highlight the capability of our model to capture accurate macroscopic dynamics over a wide range of temperatures and system sizes.

The conventional multiscale workflow typically begins by assuming a specific continuum evolution equation *a priori* and then determines the associated coefficients from experiments or atomistic simulations. Common choices for the continuum evolution equation include Fick’s laws [46], Onsager transport equations [47], and phase-field models such as the Allen-Cahn and Cahn-Hilliard equations [48–50]. For example, diffusion is typically described by Fick’s law $\partial_t c = \nabla \cdot (D \nabla c)$, where the diffusion coefficient is subsequently estimated from the mean squared displacement of molecular dynamics trajectories [51,52].

Both our workflow and the conventional multiscale workflow aim to construct effective dynamical descriptions from microscopic simulations to reduce the computational cost of tracking massive atomic degrees of freedom. However, our framework differs in several key respects. First, conventional multiscale approaches are typically formulated at a mesoscopic level, while our method learns the macroscopic dynamics. In conventional multiscale approaches, the spatially resolved field variables depend locally on microscopic coordinates. In contrast, our method explicitly targets the macroscopic dynamics. In our framework, the observables of interest depend globally on all the microscopic coordinates, resulting in macroscopic dynamics that evolve solely as a function of time rather than space. While we utilize an encoder ϕ to extract latent features $\phi(\mathbf{x}_{t,\mathcal{I}})$ from each patch,

these features are aggregated to define the state $\mathbf{z}_t = \boldsymbol{\varphi}(\mathbf{x}_t)$ for the whole large system. Our method explicitly models the dynamics of $\mathbf{z}_t$ from short-time, small-scale microscopic information, and is therefore best viewed as a macroscale modeling approach. Second, regarding the model structure, conventional multiscale approaches explicitly assume a pre-defined functional continuum evolution equation, which often limits their accuracy and ability to capture complex or non-linear phenomena. In contrast, our framework learns both the closure variables and the macroscopic dynamics directly from data, without prescribing a specific functional form. Although we employ structured architectures such as OnsagerNet, these structures serve only to enforce fundamental physical constraints, such as thermodynamic consistency, rather than to constrain the admissible functional forms of the dynamics. By parametrizing the encoder and macroscopic dynamics with neural networks, we ensure high model expressivity. Consequently, our framework offers significantly greater flexibility in discovering the underlying governing laws directly from data.

In our framework, we enforce physical consistency of the macroscopic dynamics by modeling the SDE using a stochastic OnsagerNet. The closure variables, on the other hand, are obtained through data-driven learning and may contain redundancy or lack direct physical interpretability. To reduce redundancy, one practical approach is to vary the dimension of the closure variables and identify when further increases no longer improve macroscopic prediction accuracy. In addition, physical interpretability may be improved by incorporating basic physical or symmetry constraints into the encoder, such as translation invariance for lattice systems with periodic boundary conditions, which we leave for future investigation.

A practical limitation of the present framework arises from the hierarchical upsampling procedure used to generate large-system snapshots from small-system simulations. The LOCALRELAX step is used to remove the unphysical artifacts introduced by the UPSAMPLE step. In this paper, LOCALRELAX is implemented by dividing each upsampled large-system snapshot into overlapping patches and applying short-time local dynamics evolution within each patch. This simple procedure is effective in our experiments for removing common unphysical artifacts introduced by upsampling. However, for more complex systems, this simple LOCALRELAX step may be insufficient to adequately remove the unphysical artifacts. Moreover, when applied to extremely large systems with billions of atoms, the hierarchical upsampling scheme requires many iterations, and the quality of the generated dataset D may deteriorate as the number of iterations increases. The hierarchical upsampling strategy adopted in this work represents only one possible approach, and more sophisticated strategies may be required for more complex systems. We acknowledge this as a limitation of the current method. In the future, we will explore more efficient strategies for generating large-system datasets D to mitigate this challenge.

Looking ahead, this framework holds significant promise for bridging the gap between microscopic simulations and macroscopic material behavior in complex material systems, such as high-entropy alloys [53–55] and polymer solutions [56,57]. We envision this framework as a powerful tool for

accelerating the discovery and design of advanced functional materials, with potential applications in energy storage, catalysis, and structural technologies.

ACKNOWLEDGMENTS

This research is supported by the National Research Foundation Singapore under its AI Singapore Programme (AISG Award No: AISG3-RP-2022-028), and by the Ministry of Education, Singapore, under its funding for the Research Centre of Excellence Institute for Functional Intelligent Materials (Project No. EDUNC-33-18-279-V12) and the Academic Research Fund Tier 3 Grant (Project No. MOET32024-0002). The computational work for this article was partially performed on resources of the National Supercomputing Centre, Singapore.

DATA AVAILABILITY

The data that support the findings of this article are openly available [58], embargo periods may apply.

APPENDIX A: NOTATION AND SYMBOLS

The notation and symbols are summarized in Table I.

APPENDIX B: CONTINUOUS-TIME GLAUBER DYNAMICS

The algorithm of continuous-time Glauber dynamics is summarized in Algorithm 2.

APPENDIX C: THEORETICAL ANALYSIS

Proof of Theorem 1. We divide the proof into two parts: First, we establish the closed-form expressions for the unique minimizers using variational calculus; second, we derive the error bounds based on the total variation distance.

ALGORITHM 2. Continuous-time Glauber dynamics.

Require:

$\{\sigma_i\}_{i=1}^n$: Initial spins
 β : inverse temperature
max_steps: maximum number of steps
1: Initialize step $\leftarrow 0$, kmc_time $\leftarrow 0$
2: **while** step < max_steps **do**
3: **for** $i = 1$ to n **do**
4: Compute energy differences ΔH_i by flipping spin i
5: Compute the rate $r_i = \frac{1}{1 + e^{\beta \Delta H_i}}$
6: Compute the total rate $R = \sum_{i=1}^n r_i$
7: Sample a spin j to flip with probability r_i/R
8: Sample time step $\delta t = -\frac{\ln u}{R}$, $u \sim \mathcal{U}(0, 1)$
9: kmc_time $\leftarrow$ kmc_time + δt ,
10: step $\leftarrow$ step + 1

1. Derivation of minimizers

The loss function $\mathcal{L}$ and $\mathcal{L}_p$ can be rewritten as

$$\begin{aligned} \mathcal{L}[\boldsymbol{\mu}, \boldsymbol{\Sigma}] &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_t + \delta t | \mathbf{x}_t} \\ &\quad \times [-2 \log p(\mathbf{z}_t + \delta t | \mathbf{z}_t + \boldsymbol{\mu}(\mathbf{z}_t)\delta t, \boldsymbol{\Sigma}(\mathbf{z}_t)\delta t)], \\ \mathcal{L}_p[\boldsymbol{\mu}, \boldsymbol{\Sigma}] &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t + \delta t, \mathcal{I} | \mathbf{x}_t)} \\ &\quad \times [-2 \log p(\mathbf{z}_t + \delta t, \mathcal{I} | \mathbf{z}_t + \boldsymbol{\mu}(\mathbf{z}_t)\delta t, K \boldsymbol{\Sigma}(\mathbf{z}_t)\delta t)]. \end{aligned}$$

TABLE I. Summary of key symbols and definitions.

Symbol	Meaning
$\mathbf{x}_t$	Microscopic state of the large system
$\mathcal{I}$	Index set of a randomly selected patch within the large system
δt	Time step
$x_{t,\mathcal{I}}$	Microscopic state restricted to patch $\mathcal{I}$ at time t
$x_{t+\delta t,\mathcal{I}}$	Microscopic state restricted to patch $\mathcal{I}$ at time $t + \delta t$
n_s	Number of lattice sites in the small system
n	Number of lattice sites in the large system
$\mathcal{S}_{n_s}$	Microscopic simulator that can accurately simulate the microscopic dynamics of a small system up to n_s lattice sites
D_s	Dataset of small-system snapshots
D	Training dataset constructed from D_s through hierarchical upsampling scheme
K	Number of patches used to partition the large system
φ^*	Mapping from a microscopic state to macroscopic observables
$\hat{\varphi}$	Mapping from a microscopic state to closure variables
φ	Combined encoding map, $\varphi = (\varphi^*, \hat{\varphi})$
ψ	Decoder
μ	Drift term of the latent dynamics
Σ	Diffusion term of the latent dynamics
$\mathbf{z}^*$	Macroscopic observables defined by $\mathbf{z}^* = \varphi^*(\mathbf{x})$
$\hat{\mathbf{z}}$	Closure variables defined by $\hat{\mathbf{z}} = \hat{\varphi}(\mathbf{x})$
$\mathbf{z}_t$	Latent state defined by $\mathbf{z}_t = \varphi(\mathbf{x}_t)$
$\mathbf{z}_{t+\delta t,\mathcal{I}}$	Latent state corresponding to $x_{t+\delta t,\mathcal{I}}$
$\Delta \mathbf{z}_t$	Increment of the latent state over δt , defined by $\mathbf{z}_{t+\delta t} - \mathbf{z}_t$
$\Delta \mathbf{z}_t^*$	Increment of the macroscopic observable, defined by $\mathbf{z}_{t+\delta t}^* - \mathbf{z}_t^*$
$\Delta \mathbf{z}_{t,\mathcal{I}}$	Increment of the latent state restricted to patch $\mathcal{I}$, defined by $\mathbf{z}_{t+\delta t,\mathcal{I}} - \mathbf{z}_{t,\mathcal{I}}$
$\Delta \mathbf{z}_{t,\mathcal{I}}^*$	Increment of the macroscopic observable restricted to patch $\mathcal{I}$, defined by $\mathbf{z}_{t+\delta t,\mathcal{I}}^* - \mathbf{z}_{t,\mathcal{I}}^*$

Recall $\Delta \mathbf{z}_t = \mathbf{z}_{t+\delta t} - \mathbf{z}_t$, $\Delta \mathbf{z}_{t,\mathcal{I}} = \mathbf{z}_{t+\delta t,\mathcal{I}} - \mathbf{z}_{t,\mathcal{I}}$. By introducing the precision matrix $\Lambda = \Sigma^{-1}$ and the displacement vector

$$\begin{aligned}\delta(\mathbf{z}_t, \mathbf{z}_{t+\delta t}) &:= \Delta \mathbf{z}_t - \mu(\mathbf{z}_t)\delta t \\ \delta_{\mathcal{I}}(\mathbf{z}_t, \mathbf{z}_{t+\delta t,\mathcal{I}}) &:= \Delta \mathbf{z}_{t,\mathcal{I}} - \mu(\mathbf{z}_t)\delta t,\end{aligned}$$

we can rewrite the objective as

$$\begin{aligned}\mathcal{L}[\mu, \Sigma] &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} [d \log(2\pi) + \log |\Sigma(\mathbf{z}_t)\delta t|] \\ &\quad + \delta(\mathbf{z}_t, \mathbf{z}_{t+\delta t})^T (\Sigma(\mathbf{z}_t)\delta t)^{-1} \delta(\mathbf{z}_t, \mathbf{z}_{t+\delta t}), \\ \mathcal{L}_p[\mu, \Sigma] &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t)} [d \log(2\pi) + \log |K \Sigma(\mathbf{z}_t)\delta t|] \\ &\quad + \delta_{\mathcal{I}}(\mathbf{z}_t, \mathbf{z}_{t+\delta t,\mathcal{I}})^T (K \Sigma(\mathbf{z}_t)\delta t)^{-1} \delta_{\mathcal{I}}(\mathbf{z}_t, \mathbf{z}_{t+\delta t,\mathcal{I}}).\end{aligned}$$

For further simplicity, we omit the notational dependency of δ , $\delta_{\mathcal{I}}$, μ , Λ on $\mathbf{z}_t$, $\mathbf{z}_{t+\delta t}$, $\mathbf{z}_{t+\delta t,\mathcal{I}}$. We will proceed to calculate the first-order variation and second-order variation of $\hat{\mathcal{L}}$,

$$\begin{aligned}\hat{\mathcal{L}}[\mu + \epsilon \mathbf{h}, \Lambda + \epsilon \mathbf{H}] &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} \left[d \log(2\pi) + \log(\delta t) - \log |\Lambda + \epsilon \mathbf{H}| \right. \\ &\quad \left. + \frac{1}{\delta t} (\delta - \epsilon \mathbf{h} \delta t)^T (\Lambda + \epsilon \mathbf{H}) (\delta - \epsilon \mathbf{h} \delta t) \right] \\ &= \hat{\mathcal{L}}[\mu, \Lambda] - \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} [\log |\Lambda + \epsilon \mathbf{H}| - \log |\Lambda|] \\ &\quad + \epsilon \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} \left[\frac{1}{\delta t} \delta^T \mathbf{H} \delta - 2\delta^T \Lambda \mathbf{h} \right] \\ &\quad + \frac{\epsilon^2}{2} \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} [2\delta t \mathbf{h}^T \Lambda \mathbf{h} - 4\delta^T \mathbf{H} \mathbf{h}] + o(\epsilon^2).\end{aligned}$$

Since

$$\begin{aligned}\log |\Lambda + \epsilon \mathbf{H}| &= \log |\Lambda| + \epsilon \text{tr}(\Lambda^{-1} \mathbf{H}) \\ &\quad - \frac{\epsilon^2}{2} \text{tr}(\Lambda^{-1} \mathbf{H} \Lambda^{-1} \mathbf{H}) + o(\epsilon^2),\end{aligned}$$

$$\begin{aligned}\hat{\mathcal{L}}[\mu + \epsilon \mathbf{h}, \Lambda + \epsilon \mathbf{H}] &= \hat{\mathcal{L}}[\mu, \Lambda] + \epsilon \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} \\ &\quad \times \left[-\text{tr}(\Lambda^{-1} \mathbf{H}) + \frac{1}{\delta t} \delta^T \mathbf{H} \delta - 2\delta^T \Lambda \mathbf{h} \right] \\ &\quad + \frac{\epsilon^2}{2} \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} [\text{tr}(\Lambda^{-1} \mathbf{H} \Lambda^{-1} \mathbf{H}) \\ &\quad + 2\delta t \mathbf{h}^T \Lambda \mathbf{h} - 4\delta^T \mathbf{H} \mathbf{h}] + o(\epsilon^2).\end{aligned}$$

Hence, we have the first variation $\delta \hat{\mathcal{L}}$ and second-order variation $\delta^2 \hat{\mathcal{L}}$,

$$\begin{aligned}\delta \hat{\mathcal{L}} &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} \left[-\text{tr}(\Lambda^{-1} \mathbf{H}) + \frac{1}{\delta t} \delta^T \mathbf{H} \delta - 2\delta^T \Lambda \mathbf{h} \right], \\ \delta^2 \hat{\mathcal{L}} &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} [\text{tr}(\Lambda^{-1} \mathbf{H} \Lambda^{-1} \mathbf{H}) + 2\delta t \mathbf{h}^T \Lambda \mathbf{h} - 4\delta^T \mathbf{H} \mathbf{h}].\end{aligned}$$

If μ^* , Λ^* are the minimizer of $\hat{\mathcal{L}}$, we must have $\delta \hat{\mathcal{L}} = 0$ for all the admissible $\mathbf{h}$ and $\mathbf{H}$. By a direct calculation

$$\begin{aligned}\delta \hat{\mathcal{L}} &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} \left[-\text{tr}(\Lambda^{-1} \mathbf{H}) + \frac{1}{\delta t} \delta^T \mathbf{H} \delta - 2\delta^T \Lambda \mathbf{h} \right] \\ &= \mathbb{E}_{\mathbf{x}_t, \delta t} \left[\text{tr} \left(\mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} \left[\frac{1}{\delta t} \delta \delta^T - \Lambda^{-1} \right] \mathbf{H} \right) \right] \\ &\quad + \mathbb{E}_{\mathbf{x}_t, \delta t} [\mathbb{E}_{\mathbf{x}_{t+\delta t} | \mathbf{x}_t} [-2\delta^T \Lambda] \mathbf{h}].\end{aligned}$$

The stationary condition implies

$$\begin{aligned}\mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}\left[\frac{1}{\delta t}\boldsymbol{\delta}^*(\boldsymbol{\delta}^*)^T - (\boldsymbol{\Lambda}^*)^{-1}\right] &= 0 \quad a.e., \\ \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[-2(\boldsymbol{\delta}^*)^T \boldsymbol{\Lambda}] &= \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[-2(\Delta \mathbf{z}_t - \boldsymbol{\mu}^*(\mathbf{z}_t)\delta t)^T] \boldsymbol{\Lambda}^*(\mathbf{z}_t) = 0 \quad a.e.,\end{aligned}$$

where $\boldsymbol{\delta}^*(\mathbf{z}_t, \mathbf{z}_t+\delta t) := \Delta \mathbf{z}_t - \boldsymbol{\mu}^*(\mathbf{z}_t)\delta t$. Solving these yields the unique minimizers for $\mathcal{L}$,

$$\begin{aligned}\boldsymbol{\mu}^*(\mathbf{z}_t) &= \frac{1}{\delta t} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\Delta \mathbf{z}_t] = \frac{1}{\delta t} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\mathbf{z}_t+\delta t - \mathbf{z}_t] \quad a.e., \\ \boldsymbol{\Sigma}^*(\mathbf{z}_t) &= (\boldsymbol{\Lambda}^*)^{-1}(\mathbf{z}_t) = \frac{1}{\delta t} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\boldsymbol{\delta}^*(\boldsymbol{\delta}^*)^T] \quad a.e.\end{aligned}$$

We can calculate the second variation at $\boldsymbol{\mu}^*, \boldsymbol{\Lambda}^*$,

$$\begin{aligned}\delta^2 \hat{\mathcal{L}} &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\text{tr}(\boldsymbol{\Lambda}^{-1} \mathbf{H} \boldsymbol{\Lambda}^{-1} \mathbf{H}) + 2\delta t \mathbf{h}^T \boldsymbol{\Lambda} \mathbf{h}] \\ &\quad - 4\mathbb{E}_{\mathbf{x}_t, \delta t} [\mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[(\Delta \mathbf{z}_t - \boldsymbol{\mu}^*(\mathbf{z}_t)\delta t)^T] \mathbf{H} \mathbf{h}] \\ &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\text{tr}(\boldsymbol{\Lambda}^{-1} \mathbf{H} \boldsymbol{\Lambda}^{-1} \mathbf{H}) + 2\delta t \mathbf{h}^T \boldsymbol{\Lambda} \mathbf{h}],\end{aligned}$$

$\delta^2 \hat{\mathcal{L}} > 0$ if either $\mathbf{H}$ or $\mathbf{h}$ is not equal to zero almost everywhere. Then, $\boldsymbol{\mu}^*$ and $\boldsymbol{\Lambda}^*$ are the unique minimizer of functional $\hat{\mathcal{L}}$. Equivalently, $\boldsymbol{\mu}^*$ and $\boldsymbol{\Sigma}^*$ are the unique minimizer of $\mathcal{L}$.

Similarly, we can compute the first-order variation and second-order variation of $\hat{\mathcal{L}}_p$,

$$\hat{\mathcal{L}}_p[\boldsymbol{\mu} + \epsilon \mathbf{h}, \boldsymbol{\Lambda} + \epsilon \mathbf{H}] = \hat{\mathcal{L}}_p[\boldsymbol{\mu}, \boldsymbol{\Lambda}] + \epsilon \delta \hat{\mathcal{L}}_p + \frac{\epsilon^2}{2} \delta^2 \hat{\mathcal{L}}_p + o(\epsilon^2),$$

where

$$\begin{aligned}\delta \hat{\mathcal{L}}_p &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} \\ &\quad \times \left[-\text{tr}(\boldsymbol{\Lambda}^{-1} \mathbf{H}) + \frac{1}{K\delta t} \delta_{\mathcal{I}}^T \mathbf{H} \delta_{\mathcal{I}} - \frac{2}{K} \delta_{\mathcal{I}}^T \boldsymbol{\Lambda} \mathbf{h} \right], \\ \delta^2 \hat{\mathcal{L}}_p &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} \\ &\quad \times \left[\text{tr}(\boldsymbol{\Lambda}^{-1} \mathbf{H} \boldsymbol{\Lambda}^{-1} \mathbf{H}) + \frac{2\delta t}{K} \mathbf{h}^T \boldsymbol{\Lambda} \mathbf{h} - \frac{4}{K} \delta_{\mathcal{I}}^T \mathbf{H} \mathbf{h} \right].\end{aligned}$$

If $\boldsymbol{\mu}^\dagger, \boldsymbol{\Lambda}^\dagger$ are the minimizer of $\hat{\mathcal{L}}_p$, we must have $\delta \hat{\mathcal{L}}_p = 0$ for all the admissible $\mathbf{h}$ and $\mathbf{H}$. Let $\delta_{\mathcal{I}}^\dagger(\mathbf{z}_t, \mathbf{z}_t+\delta, \mathcal{I}) := \Delta \mathbf{z}_{t, \mathcal{I}} - \boldsymbol{\mu}^\dagger(\mathbf{z}_t)\delta t$, since

$$\begin{aligned}\delta \hat{\mathcal{L}}_p &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} \\ &\quad \times \left[-\text{tr}(\boldsymbol{\Lambda}^{-1} \mathbf{H}) + \frac{1}{K\delta t} \delta_{\mathcal{I}}^T \mathbf{H} \delta_{\mathcal{I}} - \frac{2}{K} \delta_{\mathcal{I}}^T \boldsymbol{\Lambda} \mathbf{h} \right] \\ &= \mathbb{E}_{\mathbf{x}_t, \delta t} \left[\text{tr} \left(\mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} \left[\frac{1}{K\delta t} \delta_{\mathcal{I}} \delta_{\mathcal{I}}^T - \boldsymbol{\Lambda}^{-1} \right] \mathbf{H} \right) \right] \\ &\quad + \mathbb{E}_{\mathbf{x}_t, \delta t} \left[\mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} \left[-\frac{2}{K} \delta_{\mathcal{I}}^T \boldsymbol{\Lambda} \right] \mathbf{h} \right],\end{aligned}$$

we have

$$\begin{aligned}\mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} \left[-\frac{2}{K} (\delta_{\mathcal{I}}^\dagger)^T \boldsymbol{\Lambda}^\dagger \right] \\ = \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} \left[-\frac{2}{K} (\Delta \mathbf{z}_{t, \mathcal{I}} - \boldsymbol{\mu}^\dagger(\mathbf{z}_t)\delta t)^T \boldsymbol{\Lambda}^\dagger(\mathbf{z}_t) \right] \\ = 0 \quad a.e. \\ \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} \left[\frac{1}{K\delta t} \delta_{\mathcal{I}}^\dagger (\delta_{\mathcal{I}}^\dagger)^T - (\boldsymbol{\Lambda}^\dagger)^{-1} \right] = 0 \quad a.e.\end{aligned}$$

Hence,

$$\begin{aligned}\boldsymbol{\mu}^\dagger(\mathbf{z}_t) &= \frac{1}{\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} [\Delta \mathbf{z}_{t, \mathcal{I}}] \\ &= \frac{1}{\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} [\mathbf{z}_t+\delta t, \mathcal{I} - \mathbf{z}_t] \quad a.e., \\ \boldsymbol{\Sigma}^\dagger(\mathbf{z}_t) &= (\boldsymbol{\Lambda}^\dagger)^{-1}(\mathbf{z}_t) = \frac{1}{K\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} [\delta_{\mathcal{I}}^\dagger (\delta_{\mathcal{I}}^\dagger)^T] \quad a.e.\end{aligned}$$

We can calculate the second variation at $\boldsymbol{\mu}^\dagger, \boldsymbol{\Lambda}^\dagger$,

$$\begin{aligned}\delta^2 \hat{\mathcal{L}}_p &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} \left[\text{tr}(\boldsymbol{\Lambda}^{-1} \mathbf{H} \boldsymbol{\Lambda}^{-1} \mathbf{H}) + \frac{2\delta t}{K} \mathbf{h}^T \boldsymbol{\Lambda}^\dagger \mathbf{h} \right] \\ &\quad - \frac{4}{K} \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} [\delta_{\mathcal{I}}^T \mathbf{H} \mathbf{h}] \\ &= \mathbb{E}_{\mathbf{x}_t, \delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{q(\mathbf{x}_t+\delta t, \mathcal{I}|\mathbf{x}_t)} \left[\text{tr}(\boldsymbol{\Lambda}^{-1} \mathbf{H} \boldsymbol{\Lambda}^{-1} \mathbf{H}) + \frac{2\delta t}{K} \mathbf{h}^T \boldsymbol{\Lambda}^\dagger \mathbf{h} \right].\end{aligned}$$

2. Error analysis

We now bound the difference between the estimators. To bound $\|\boldsymbol{\mu}^* - \boldsymbol{\mu}^\dagger\|_\infty$, we use the property of total variation distance. For any random variable X bounded by B , we have

$$\|\mathbb{E}_q[X] - \mathbb{E}_{\hat{q}}[X]\| \leq 2B\delta_{\text{TV}}(q, \hat{q}).$$

Next, since the encoder $\boldsymbol{\varphi}$ is uniformly bounded by M , it follows that $\|\Delta \mathbf{z}_{t, \mathcal{I}}\|_\infty \leq 2M$. We therefore obtain

$$\begin{aligned}\|\boldsymbol{\mu}^\dagger(\mathbf{z}_t) - \boldsymbol{\mu}^*(\mathbf{z}_t)\| &= \left\| \frac{1}{\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\Delta \mathbf{z}_{t, \mathcal{I}}] - \frac{1}{\delta t} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\Delta \mathbf{z}_t] \right\| \\ &\leq \left\| \frac{1}{\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\Delta \mathbf{z}_{t, \mathcal{I}}] - \frac{1}{\delta t} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t} \left[\frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t, \mathcal{I}} \right] \right\| \\ &\quad + \frac{1}{\delta t} \left\| \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t} \left[\Delta \mathbf{z}_t - \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t, \mathcal{I}} \right] \right\| \\ &= \left\| \frac{1}{\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\Delta \mathbf{z}_{t, \mathcal{I}}] - \frac{1}{\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t, \mathcal{I}}] \right\| \\ &\quad + \frac{1}{\delta t} \left\| \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t} \left[\Delta \mathbf{z}_t - \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t, \mathcal{I}} \right] \right\|.\end{aligned}$$

By the definition of the closure variables, we have $\Delta \hat{\mathbf{z}}_t = \frac{1}{K} \sum_{\mathcal{I}} \Delta \hat{\mathbf{z}}_{t, \mathcal{I}}$. By the assumption

$$\left\| \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t} [\Delta \mathbf{z}_t^* - \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t, \mathcal{I}}^*] \right\| \leq C_2 \delta t^2,$$

then

$$\begin{aligned}\frac{1}{\delta t} \left\| \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t} \left[\Delta \mathbf{z}_t - \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t, \mathcal{I}} \right] \right\| \\ = \frac{1}{\delta t} \left\| \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t} \left[\Delta \mathbf{z}_t^* - \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t, \mathcal{I}}^* \right] \right\| \leq C_2 \delta t.\end{aligned}$$

By the property of total variation distance,

$$\begin{aligned} & \left\| \frac{1}{\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\Delta \mathbf{z}_{t,\mathcal{I}}] - \frac{1}{\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}] \right\| \\ & \leq \frac{1}{\delta t} \cdot 4M \cdot \delta_{\text{TV}}(q, \hat{q}) \leq 4MC_1 \delta t. \end{aligned}$$

Hence,

$$\|\boldsymbol{\mu}^* - \boldsymbol{\mu}^\dagger\|_\infty \leq (4MC_1 + C_2) \delta t.$$

We now bound $\|\boldsymbol{\Sigma}^\dagger - \boldsymbol{\Sigma}^*\|_\infty$. We define

$$\begin{aligned} \boldsymbol{\delta}_{\mathcal{I}}^*(\mathbf{z}_t, \mathbf{z}_{t+\delta t, \mathcal{I}}) &= \Delta \mathbf{z}_{t,\mathcal{I}} - \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\Delta \mathbf{z}_{t,\mathcal{I}}] \\ &= \Delta \mathbf{z}_{t,\mathcal{I}} - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}]. \end{aligned}$$

Since the encoder is uniformly bounded by M , we have

$$\begin{aligned} \|\boldsymbol{\delta}_{\mathcal{I}}^*\|_\infty &\leq 4M, \\ \|\boldsymbol{\delta}_{\mathcal{I}}^*(\boldsymbol{\delta}_{\mathcal{I}}^*)^T\|_\infty &\leq (4M)^2 = 16M^2. \end{aligned}$$

By making use of the following decomposition of $\boldsymbol{\delta}^*$,

$$\begin{aligned} \boldsymbol{\delta}^* &= \Delta \mathbf{z}_t - \boldsymbol{\mu}^*(\mathbf{z}_t) \delta t \\ &= \frac{1}{K} \sum_{\mathcal{I}} \boldsymbol{\delta}_{\mathcal{I}}^* + \left(\Delta \mathbf{z}_t - \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t,\mathcal{I}} \right) \\ &\quad - \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t} \left[\Delta \mathbf{z}_t - \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t,\mathcal{I}} \right] \end{aligned}$$

we have

$$\begin{aligned} & \|\boldsymbol{\Sigma}^\dagger(\mathbf{z}_t) - \boldsymbol{\Sigma}^*(\mathbf{z}_t)\| \\ &= \left\| \frac{1}{K\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T] - \frac{1}{\delta t} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\boldsymbol{\delta}^*(\boldsymbol{\delta}^*)^T] \right\| \\ &\leq \left\| \frac{1}{K\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T] - \frac{1}{K^2\delta t} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t} \left[\sum_{\mathcal{I}, \mathcal{J}} \boldsymbol{\delta}_{\mathcal{I}}^*(\boldsymbol{\delta}_{\mathcal{J}}^*)^T \right] \right\| \\ &\quad + 4 \left\| \frac{1}{K} \sum_{\mathcal{I}} \boldsymbol{\delta}_{\mathcal{I}}^* \right\|_\infty \cdot \left\| \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t} \left[\Delta \mathbf{z}_t - \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t,\mathcal{I}} \right] \right\| \\ &\quad + \mathcal{O}(\delta t^3) \\ &\leq \left\| \frac{1}{K\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T] - \frac{1}{K\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\boldsymbol{\delta}_{\mathcal{I}}^*(\boldsymbol{\delta}_{\mathcal{I}}^*)^T] \right\| \\ &\quad + \left\| \frac{1}{K^2\delta t} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t} \left[\sum_{\mathcal{I} \neq \mathcal{J}} \boldsymbol{\delta}_{\mathcal{I}}^*(\boldsymbol{\delta}_{\mathcal{J}}^*)^T \right] \right\| \\ &\quad + 16MC_2\delta t + \mathcal{O}(\delta t^3) \end{aligned}$$

Using the decomposition $\boldsymbol{\delta}_{\mathcal{I}}^* = \boldsymbol{\delta}_{\mathcal{I}}^\dagger + (\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])$, we can decompose $\boldsymbol{\delta}_{\mathcal{I}}^*(\boldsymbol{\delta}_{\mathcal{I}}^*)^T$,

$$\begin{aligned} \boldsymbol{\delta}_{\mathcal{I}}^*(\boldsymbol{\delta}_{\mathcal{I}}^*)^T &= \boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T + \boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])^T \\ &\quad + (\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T \\ &\quad + (\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])(\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])^T. \end{aligned}$$

We have

$$\begin{aligned} & \left\| \frac{1}{K\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T] - \frac{1}{K\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\boldsymbol{\delta}_{\mathcal{I}}^*(\boldsymbol{\delta}_{\mathcal{I}}^*)^T] \right\| \\ & \leq \frac{1}{K\delta t} \|\mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T] - \mathbb{E}_{\mathcal{I}} \mathbb{E}_{\hat{q}}[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T]\| \\ & \quad + \frac{2}{K\delta t} \|\mathbb{E}_{\mathcal{I}} \mathbb{E}_{\hat{q}}[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])^T]\| \\ & \quad + \frac{1}{K\delta t} \|\mathbb{E}_{\mathcal{I}}[(\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])(\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])^T]\|. \end{aligned}$$

We can bound the first term by

$$\begin{aligned} & \frac{1}{K\delta t} \|\mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T] - \mathbb{E}_{\mathcal{I}} \mathbb{E}_{\hat{q}}[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T]\| \\ & \leq \frac{2 \cdot (4M)^2}{K\delta t} \delta_{\text{TV}}(q, \hat{q}) \leq \frac{32M^2C_1}{K} \delta t. \end{aligned}$$

Next, we bound the second term

$$\begin{aligned} & \frac{2}{K\delta t} \|\mathbb{E}_{\mathcal{I}} \mathbb{E}_{\hat{q}}[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])^T]\| \\ & \leq \frac{2}{K\delta t} \|\boldsymbol{\delta}_{\mathcal{I}}^\dagger\|_\infty \cdot \|\mathbb{E}_{\mathcal{I}} \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}] - \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\Delta \mathbf{z}_{t,\mathcal{I}}]\| \\ & \leq \frac{8M}{K\delta t} \cdot 4M \cdot \delta_{\text{TV}}(q, \hat{q}) \leq \frac{32M^2C_1}{K} \delta t. \end{aligned}$$

We can also bound the third term

$$\begin{aligned} & \frac{1}{K\delta t} \|\mathbb{E}_{\mathcal{I}}[(\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])(\boldsymbol{\mu}^\dagger - \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}])^T]\| \\ & \leq \frac{1}{K\delta t} \|\mathbb{E}_{\mathcal{I}} \mathbb{E}_{\hat{q}}[\Delta \mathbf{z}_{t,\mathcal{I}}] - \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\Delta \mathbf{z}_{t,\mathcal{I}}]\|_\infty^2 \\ & \leq \frac{1}{K\delta t} \delta_{\text{TV}}(q, \hat{q})^2 \leq \frac{C_2^2}{K} \delta t^3. \end{aligned}$$

Combining the above bounds, we conclude that

$$\begin{aligned} & \left\| \frac{1}{K\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_q[\boldsymbol{\delta}_{\mathcal{I}}^\dagger(\boldsymbol{\delta}_{\mathcal{I}}^\dagger)^T] - \frac{1}{K\delta t} \mathbb{E}_{\mathcal{I}} \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\boldsymbol{\delta}_{\mathcal{I}}^*(\boldsymbol{\delta}_{\mathcal{I}}^*)^T] \right\| \\ & \leq \frac{64M^2C_1}{K} \delta t + \mathcal{O}(\delta t^3). \end{aligned}$$

By the assumption that increments $\mathbf{z}_{t+\delta t, \mathcal{I}} - \mathbf{z}_t$ and $\mathbf{z}_{t+\delta t, \mathcal{J}} - \mathbf{z}_t$ are independent for disjoint $\mathcal{I} \neq \mathcal{J}$ given $\mathbf{x}_t$, the cross-covariance terms vanish. Specifically,

$$\mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\boldsymbol{\delta}_{\mathcal{I}}^*(\boldsymbol{\delta}_{\mathcal{J}}^*)^T] = \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\boldsymbol{\delta}_{\mathcal{I}}^*] \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[(\boldsymbol{\delta}_{\mathcal{J}}^*)^T].$$

Since

$$\mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\boldsymbol{\delta}_{\mathcal{I}}^*] = \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\Delta \mathbf{z}_{t,\mathcal{I}}] - \mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\Delta \mathbf{z}_{t,\mathcal{I}}] = 0$$

we have

$$\mathbb{E}_{\mathbf{x}_t+\delta t|\mathbf{x}_t}[\boldsymbol{\delta}_{\mathcal{I}}^*(\boldsymbol{\delta}_{\mathcal{J}}^*)^T] = 0,$$

which yields

$$\|\boldsymbol{\Sigma}^\dagger(\mathbf{z}_t) - \boldsymbol{\Sigma}^*(\mathbf{z}_t)\| \leq \frac{64M^2C_1}{K} \delta t + 16MC_2\delta t + \mathcal{O}(\delta t^3).$$

■

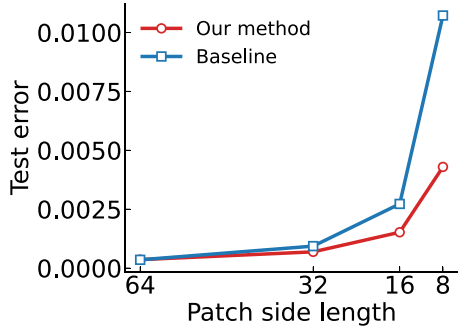


FIG. 8. Results on the Curie-Weiss model, where the test error is plotted as a function of n_s . The test error is the mean relative error of the mean macroscopic observables between ground-truth and predicted trajectories.

APPENDIX D: ADDITIONAL EXPERIMENTAL RESULTS

1. Curie-Weiss model

As an additional robustness check, we consider the Curie-Weiss model, which does not strictly satisfy the assumptions of our framework. The Curie-Weiss model is a mean-field system with nonlocal interactions [59,60]. Each spin interacts with all others through a global coupling. The Hamiltonian of the Curie-Weiss model with n spins is given by

$$H_{n,h}(\sigma) = -\frac{J}{2n} \sum_{i,j=1}^n \sigma_i \sigma_j - h \sum_{i=1}^n \sigma_i,$$

where $\sigma_i \in \{-1, 1\}$ and h denotes the external magnetic field. The Hamiltonian can also be expressed as a function of the magnetization $M = \sum_i \sigma_i/n$,

$$H_{n,h}(\sigma) = -n \left[\frac{J}{2} \left(\frac{\sum_i \sigma_i}{n} \right)^2 + h \frac{\sum_i \sigma_i}{n} \right].$$

Hence, the microscopic dynamics can be fully characterized by the magnetization. We choose the magnetization as the macroscopic observable of interest since the magnetization is closed by itself, and no additional closure variables are needed. As in the Ising model, we adopt the continuous-time Glauber dynamics as the microscopic dynamics.

The Curie-Weiss model does not fit directly into our framework, because it is a mean-field model without any explicit spatial structure. To handle this, we imagine the spins are arranged on a square lattice.

We repeat the ablation study described in Sec. III B with $h = 0.1$ and $T = 1.1 > T_c = 1$, using the same UPSAMPLE and LOCALRELAX steps. Figures 8 and 9 report the corresponding results. We observe that our method consistently outperforms the baseline when $n_s < n$. Moreover, the trajectory statistics produced by our method closely match those of the ground truth, whereas the baseline exhibits much larger variability. Although the Curie-Weiss model does not strictly satisfy the locality assumptions underlying our framework, our method continues to yield reasonable macroscopic predictions. These observations suggest empirical robustness of our framework.

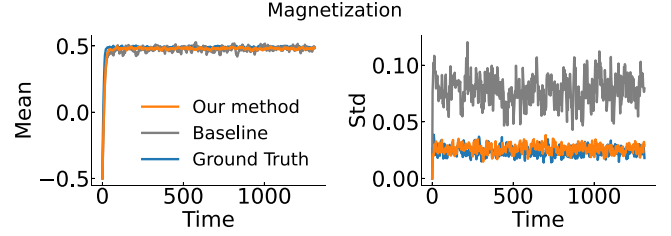


FIG. 9. Results on the Curie-Weiss model with $n_s = 16^2$. Mean and standard deviation of the magnetization are estimated from 20 trajectories per method.

2. Linear chain with boundary driving

Our method aims to learn the macroscopic dynamics of a large system using only data generated from small-system simulations. Consider a small system with N particles and a large system with KN particles governed by the same microscopic dynamics. In the modified SDE loss, we multiply the noise covariance by a factor K , which is equivalent to scaling the diffusion coefficient by $1/\sqrt{K}$.

A natural question then arises: Instead of modifying the loss, could one simply learn the macroscopic dynamics of the small system using the conventional method, and then rescale the diffusion term by $1/\sqrt{K}$ at prediction time to obtain the macroscopic dynamics of the large system? The answer is, in general, no.

The reason is that, for a given macroscopic observable, increasing the system size may change not only the diffusion level but also the macroscopic drift. We illustrate this point using an analytical stochastic differential equation (SDE) example.

Specifically, to demonstrate why training on a small system and merely rescaling the diffusion is insufficient, we consider a one-dimensional chain of N particles with linear on-site friction $\gamma > 0$ and nearest-neighbor coupling $\kappa > 0$. The system is driven by a constant external force F applied only to the first particle. The microscopic dynamics for the displacement $X_i(t)$ are given by

$$dX_1 = [-\gamma X_1 + \kappa(X_2 - X_1) + F] dt + \sigma dW_1,$$

$$dX_i = [-\gamma X_i + \kappa(X_{i-1} - 2X_i + X_{i+1})] dt + \sigma dW_i,$$

$$i = 2, \dots, N-1,$$

$$dX_N = [-\gamma X_N + \kappa(X_{N-1} - X_N)] dt + \sigma dW_N,$$

where $\{W_i\}_{i=1}^N$ are independent Brownian motions.

Let the macroscopic observable be the mean displacement $m_N(t) = \frac{1}{N} \sum_{i=1}^N X_i(t)$. Summing the microscopic equations over i yields the closed macroscopic dynamics,

$$dm_N(t) = \underbrace{\left[-\gamma m_N(t) + \frac{F}{N} \right]}_{\text{drift } f_N(m)} dt + \underbrace{\frac{\sigma}{\sqrt{N}}}_{\text{diffusion } g_N} dB(t),$$

$$dB(t) = \frac{1}{\sqrt{N}} \sum_{i=1}^N dW_i(t).$$

Now consider a small system of size N and a large system of size KN , where the same boundary driving is applied to

TABLE II. Comparison of theoretical and learned parameters for the reduced macroscopic SDE.

Parameter	Small system ($N = 10$)		Large system ($KN = 100$)	
	Learned (Conventional)	Theoretical	Learned (Ours)	
Slope a ($-\gamma$)	-0.10	-0.1047	-0.10	-0.0929
Bias b (F/N)	1.50	1.5002	0.15	0.1419
Diffusion c ($\sigma/\sqrt{N}$)	0.316	0.3122	0.100	0.1000

a single particle. Even though the diffusion term rescales as $1/\sqrt{K}$, the drift terms are also different,

$$f_{KN}(m) = -\gamma m + \frac{F}{KN} \neq -\gamma m + \frac{F}{N} = f_N(m).$$

This shows that learning from a small system and only adjusting the diffusion is insufficient to recover the correct macroscopic drift in the large system.

We also demonstrate the difference in the learned drift through experiments. We set $N = 10$, $K = 10$, $F = 15$, $\sigma = 1$, $\gamma = 0.1$, and $\kappa = 1$. We compare two training strategies:

(1) Conventional method. We simulate a small system of size N to obtain pairs $(x_t, x_{t+\delta t})$ from the full microscopic dynamics, and train an SDE model using the standard loss $\mathcal{L}$.

(2) Our method. We simulate a system of size KN , partition it into K local patches, and generate partial updates $\mathbf{x}_{t+\delta t, \mathcal{I}}$ using our partial evolution scheme. We then train the macroscopic SDE using our modified loss $\mathcal{L}_p$.

Since the true macroscopic drift is linear and the diffusion is constant, we restrict the drift network to be linear and the diffusion network to output a constant. Specifically, we fit

$$dm(t) = (a m(t) + b) dt + c dB(t),$$

so that each method estimates only three parameters (a , b , c). The results are summarized in Table II. We observe that the conventional method accurately recovers the macroscopic dynamics of the small system, whereas our method accurately recovers the macroscopic dynamics of the large system.

3. Additional experiment of Ising model

We next validate the above observation using the two-dimensional Ising model from the main paper. We take the small system to be a 16×16 Ising model and the large system to be 64×64 . We consider temperature $T = 2.25$ and external field $h = 0$, where the temperature is close to the critical temperature.

We compare the following three settings:

(1) $\mathbf{x}_t$ is sampled from snapshots of small-system trajectories, and $\mathbf{x}_{t+\delta t}$ is obtained by fully evolving the small system for a short time. The model is trained with the conventional SDE loss $\mathcal{L}$.

(2) We use the same SDE as in (i). We scale the diffusion term of the SDE by $1/\sqrt{16} = 1/4$ during prediction.

(3) $\mathbf{x}_t$ is sampled from snapshots of large-system trajectories, and $\mathbf{x}_{t+\delta t, \mathcal{I}}$ is generated by the partial evolution scheme. The model is trained with our modified SDE loss.

The results are shown in Fig. 10. Since $T = 2.25$ is close to the critical temperature, the Ising model undergoes

frequent magnetization reversals, and the magnetization therefore changes sign repeatedly over time.

We observe that the SDE trained under setting (i) accurately reproduces the equilibrium magnetization distribution of the 16×16 Ising model. However, when the same SDE is used for prediction with the diffusion term scaled by $1/4$ as in setting (ii), the predicted magnetization no longer exhibits magnetization reversals. As a consequence, the resulting equilibrium distribution deviates significantly from that of the 64×64 Ising model. In contrast, under setting (iii), our method successfully reproduces the equilibrium magnetization distribution of the 64×64 Ising model.

The failure of setting (ii) demonstrates that simply rescaling the diffusion term is insufficient for extrapolating macroscopic dynamics from small to large systems. This is because of the difference in the drift terms between the small-system macroscopic dynamics and the large-system macroscopic dynamics. This shows that learning macroscopic dynamics from small-system simulations alone is nontrivial, highlighting the necessity and importance of our method.

4. Ablation study

The conventional SDE loss $\mathcal{L}$ in Eq. (6) and our modified SDE loss $\mathcal{L}_p$ in Eq. (7) involve two data distributions: the distribution of input states $\mathbf{x}_t$ and the distribution of short-time evolved states $\mathbf{x}_{t+\delta t}$ or $\mathbf{x}_{t+\delta t, \mathcal{I}}$. Accordingly, when relying on small-system simulations, two corresponding sources of distribution shift arise.

The first source arises from a mismatch in the distribution of $\mathbf{x}_t$, as configurations constructed from small-system simulations generally differ from snapshots sampled from large-system trajectories. To mitigate this effect, we introduce a hierarchical upsampling scheme to construct large-system configurations from small-system snapshots.

The second source of distribution shift concerns the distribution of the short-time evolved states. In particular, the partial evolution scheme induces a mismatch between the distribution of $\mathbf{x}_{t+\delta t, \mathcal{I}}$ and that of $\mathbf{x}_{t+\delta t}$ obtained by evolving the full microscopic system. To address this effect, we introduce the modified SDE loss $\mathcal{L}_p$ together with Theorem 1, which provides a theoretical justification for correcting the statistical bias introduced by partial evolution. Importantly, Theorem 1 applies specifically to the distribution shift in $\mathbf{x}_{t+\delta t, \mathcal{I}}$, while the distribution shift in $\mathbf{x}_t$ is shared by both the conventional loss $\mathcal{L}$ and the modified loss $\mathcal{L}_p$. This learning formulation is general and applicable across different systems.

To disentangle these two sources of distribution shift, we perform an ablation study. We consider three ways of constructing the input dataset $\mathbf{x}_t$: (i) snapshots sampled

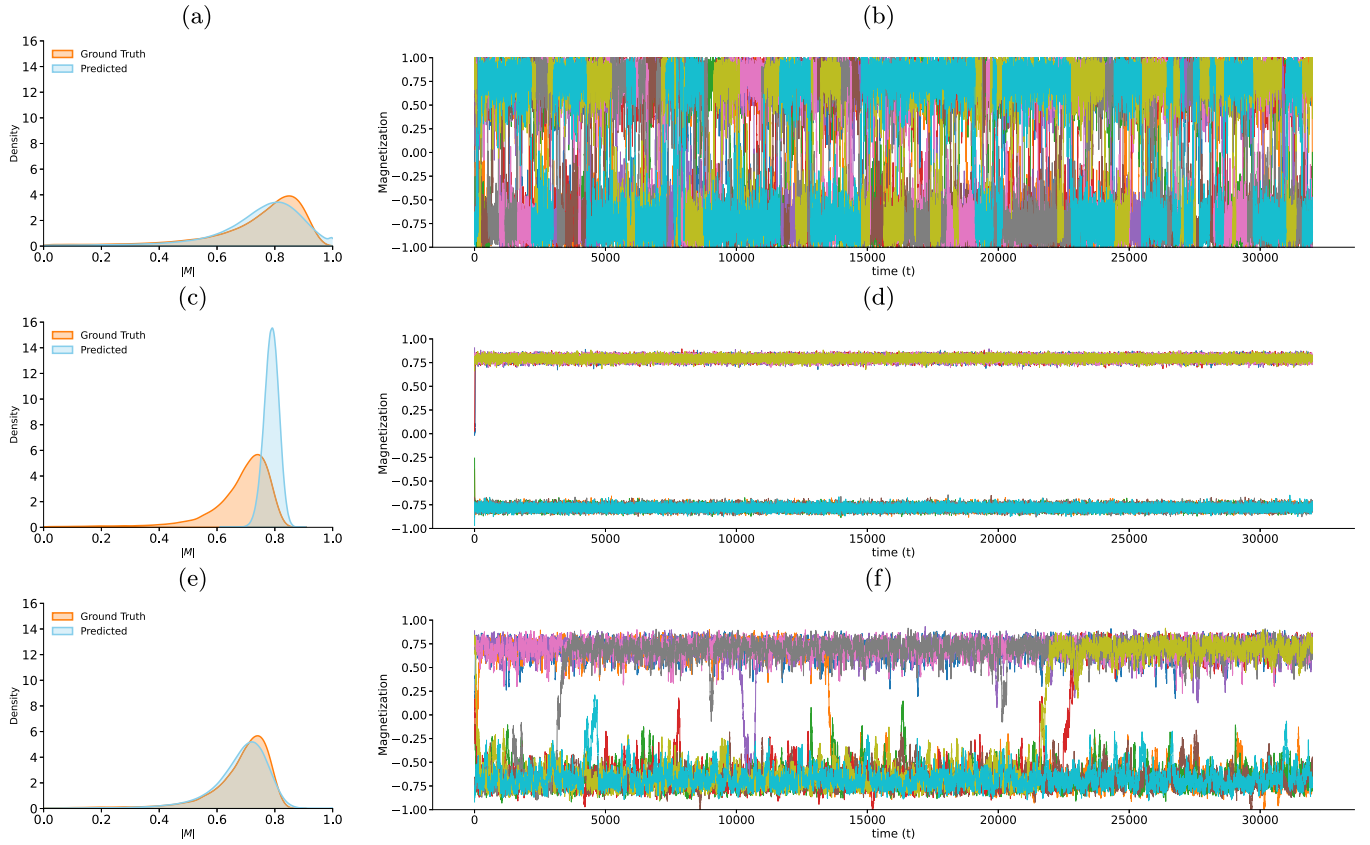


FIG. 10. Comparison of magnetization statistics and dynamics. Each row corresponds to settings (i)–(iii) from top to bottom. The first column shows the predicted equilibrium magnetization distribution together with the ground truth. In the first row, the ground truth is obtained from direct simulations of a 16×16 Ising system, while in the second and third rows it is obtained from direct simulations of a 64×64 system. The second column shows representative predicted trajectories of the magnetization dynamics.

from large-system trajectories; (ii) snapshots obtained by naive upsampling from small-system trajectories without LOCALRELAX; and (iii) snapshots generated by the hierarchical upsampling scheme.

Correspondingly, we consider three choices for generating the short-time evolved states and the associated training loss: (i) Conventional method: full microscopic evolution with the conventional loss $\mathcal{L}$; (ii) Baseline: partial evolution with the conventional loss $\mathcal{L}$; and (iii) Our method: partial evolution with the proposed loss $\mathcal{L}_p$.

The experimental settings are summarized in Table III. The large system is a 64×64 Ising model, and the small system is

a 16×16 Ising model. We use temperature $T = 2.25$ and external field $h = 0$. For each setting, we train the macroscopic dynamics model and perform long-term prediction. We then compare the predicted equilibrium magnetization distribution with the ground truth. The corresponding results are shown in Fig. 11.

The results for the second row of Table III are not shown. When snapshots obtained by naive upsampling from small-system snapshots are used, the distribution shift in $\mathbf{x}_t$ is too large for the SDE dynamics to be learned. As a result, the long-term predictions diverge and produce NaN values.

TABLE III. Summary of experimental settings used in the ablation study. Each table entry corresponds to a specific experimental setting, labeled by a letter that matches the subfigure label in Fig. 11.

Dataset of $\mathbf{x}_t$	Conventional method $\mathbf{x}_{t+\delta t}$: Full evolution Training: loss $\mathcal{L}$	Baseline $\mathbf{x}_{t+\delta t, \mathcal{I}}$: Partial evolution scheme Training: loss $\mathcal{L}$	Our method $\mathbf{x}_{t+\delta t, \mathcal{I}}$: Partial evolution scheme Training loss $\mathcal{L}_p$
64×64 Ising trajectory snapshots	(a)	(b)	(c)
Upsample (no LOCALRELAX)			
Upsampled from 16×16 snapshots			
Hierarchical upsampling scheme generated from 16×16 snapshots	(d)	(e)	(f)

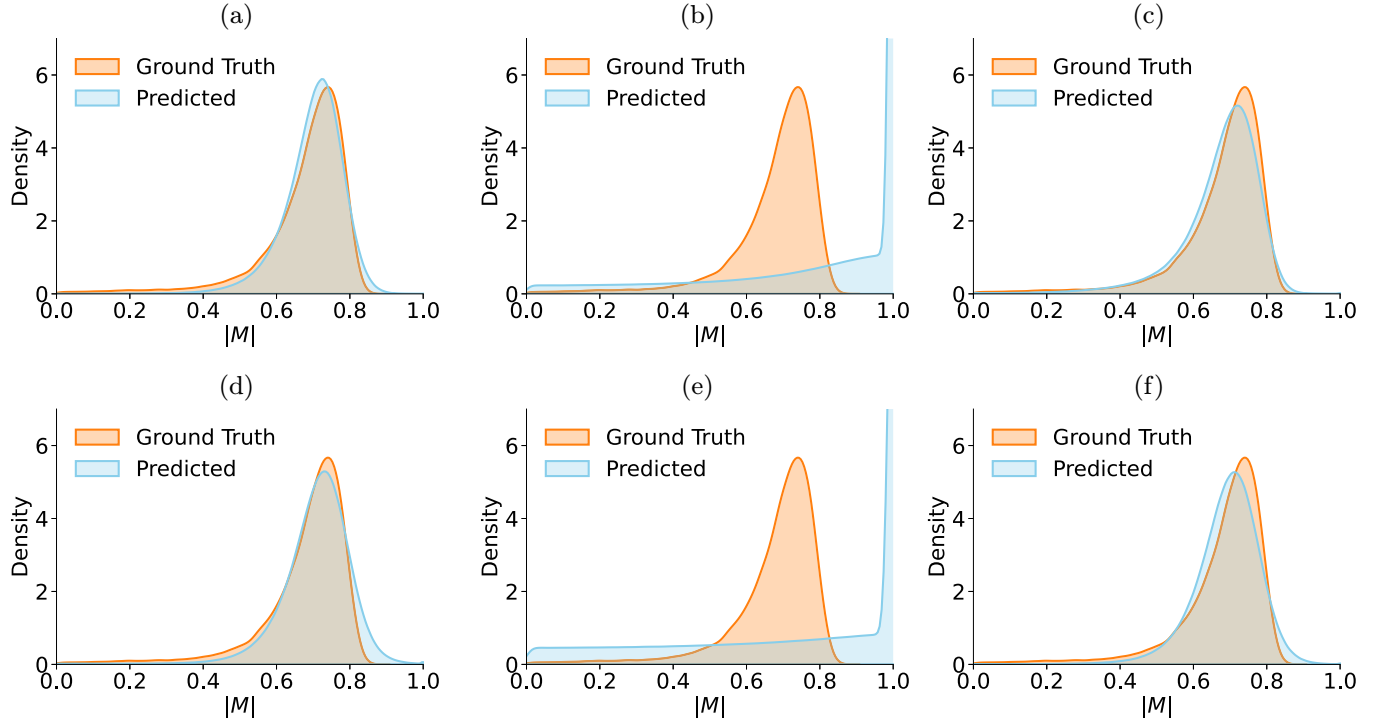


FIG. 11. Equilibrium magnetization distributions obtained by long-term prediction of the learned macroscopic dynamics under different experimental settings. Panels (a)–(c) correspond to the first row of Table III, where $\mathbf{x}_t$ is sampled from large-system trajectories. Panels (d)–(f) correspond to the third row of Table III, where $\mathbf{x}_t$ is generated using the hierarchical upsampling scheme. From left to right, each column shows results obtained using full microscopic evolution with loss $\mathcal{L}$ (conventional method), partial evolution with loss $\mathcal{L}$ (baseline), and partial evolution with the proposed loss $\mathcal{L}_p$ (our method).

From the ablation study, we observe the following:

(1) When the distribution shift in $\mathbf{x}_t$ is too large, neither full evolution nor partial evolution can learn the macroscopic dynamics well, regardless of the training loss used.

(2) The LOCALRELAX step helps remove unphysical artifacts introduced by the UPSAMPLE step. By comparing the first and second rows of Fig. 11, we see that the hierarchical upsampling scheme produces a more reasonable distribution of $\mathbf{x}_t$ than naive upsampling.

(3) Comparing each column, regardless of how $\mathbf{x}_t$ is generated, the proposed loss $\mathcal{L}_p$ consistently mitigates the error introduced by partial evolution compared to the conventional loss.

In summary, the hierarchical upsampling scheme helps produce a more reasonable distribution of $\mathbf{x}_t$ and mitigates the distribution shift in $\mathbf{x}_t$, while the modified SDE loss acts solely to correct the additional variance introduced by the partial evolution scheme.

APPENDIX E: ADDITIONAL PROOFS

We will show that the stochastic predator-prey system exactly satisfies the condition of Theorem 1.

Proof. The spatial domain $[0,1]$ is discretized into $n = 200$ uniform grids with $x_i = (i - \frac{1}{2})\Delta x$, $\Delta x = 1/200$, $1 \leq i \leq 200$. Let

$$\mathbf{u}_t = (u(x_1, t), \dots, u(x_{200}, t)),$$

$$\mathbf{v}_t = (v(x_1, t), \dots, v(x_{200}, t)),$$

and treat $(\mathbf{u}_t, \mathbf{v}_t) \in \mathbb{R}^{400}$ as the microscopic state. We approximate the spatial derivatives using finite differences. For $2 \leq i \leq 199$,

$$\frac{\partial^2 u}{\partial x^2}(x_i, t) \approx \frac{u(x_{i+1}, t) - 2u(x_i, t) + u(x_{i-1}, t))}{\Delta x^2},$$

and at the boundaries,

$$\begin{aligned} \frac{\partial^2 u}{\partial x^2}(x_1, t) &\approx \frac{u(x_2, t) - u(x_1, t)}{\Delta x^2}, \\ \frac{\partial^2 u}{\partial x^2}(x_{200}, t) &\approx \frac{u(x_{199}, t) - u(x_{200}, t)}{\Delta x^2}. \end{aligned}$$

The same discretization is used for v .

Define $h_u(u, v) = u(1 - u - v)$ and $h_v(u, v) = av(u - b)$, and let $h_u(\mathbf{u}, \mathbf{v})$ and $h_v(\mathbf{u}, \mathbf{v})$ denote their elementwise application. The semi-discrete system can be written as

$$\begin{aligned} \frac{d\mathbf{u}}{dt} &= h_u(\mathbf{u}, \mathbf{v}) + c\mathbf{A}\mathbf{u} + \sigma_u d\mathbf{B}_t^u, \\ \frac{d\mathbf{v}}{dt} &= h_v(\mathbf{u}, \mathbf{v}) + \mathbf{A}\mathbf{v} + \sigma_v d\mathbf{B}_t^v, \end{aligned}$$

where $\mathbf{B}_t^u, \mathbf{B}_t^v \in \mathbb{R}^{200}$ are independent standard Brownian motions, and the matrix $\mathbf{A} \in \mathbb{R}^{200 \times 200}$ is given by

$$\mathbf{A} = \frac{1}{\Delta x^2} \begin{pmatrix} -1 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \ddots & \vdots \\ 0 & 1 & -2 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & 1 & -1 \end{pmatrix}.$$

For time discretization, we apply the Euler-Maruyama scheme,

$$\begin{aligned} \mathbf{u}_{t+\delta t} &= \mathbf{u}_t + \delta t(h_u(\mathbf{u}_t, \mathbf{v}_t) + c\mathbf{A}\mathbf{u}_t) + \sigma_u \delta \mathbf{B}_t^u, \\ \mathbf{v}_{t+\delta t} &= \mathbf{v}_t + \delta t(h_v(\mathbf{u}_t, \mathbf{v}_t) + \mathbf{A}\mathbf{v}_t) + \sigma_v \delta \mathbf{B}_t^v, \end{aligned}$$

where $\delta \mathbf{B}_t^u$ and $\delta \mathbf{B}_t^v$ are independent Gaussian random vectors distributed as $\mathcal{N}(\mathbf{0}, \delta t \mathbf{I}_{200})$.

Step 1: $q(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t) = \hat{q}(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t)$. The conditional distribution of $\mathbf{x}_{t+\delta t} = (\mathbf{u}_{t+\delta t}, \mathbf{v}_{t+\delta t})$ given $\mathbf{x}_t = (\mathbf{u}_t, \mathbf{v}_t)$ is

$$\begin{aligned} \mathbf{x}_{t+\delta t} | \mathbf{x}_t &\sim \mathcal{N}\left(\begin{pmatrix} \mathbf{u}_t \\ \mathbf{v}_t \end{pmatrix} + \delta t \begin{pmatrix} h_u(\mathbf{u}_t, \mathbf{v}_t) + c\mathbf{A}\mathbf{u}_t \\ h_v(\mathbf{u}_t, \mathbf{v}_t) + \mathbf{A}\mathbf{v}_t \end{pmatrix}, \right. \\ &\quad \left. \delta t \begin{pmatrix} \sigma_u^2 \mathbf{I}_{200} & \mathbf{0} \\ \mathbf{0} & \sigma_v^2 \mathbf{I}_{200} \end{pmatrix}\right), \end{aligned}$$

which is obtained by fully evolving the microscopic state over a time interval of length δt using the Euler-Maruyama scheme. Then the conditional distribution $\hat{q}(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t)$ obtained by restricting $\mathbf{x}_{t+\delta t}$ to a local spatial patch $\mathcal{I} = \{k, \dots, l\} \subset \{1, \dots, 200\}$ is given by the marginal of $q(\mathbf{x}_{t+\delta t} | \mathbf{x}_t)$,

$$\begin{aligned} \hat{q}(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t) &= \mathcal{N}\left(\begin{pmatrix} \mathbf{u}_{t, \mathcal{I}} \\ \mathbf{v}_{t, \mathcal{I}} \end{pmatrix} + \delta t \begin{pmatrix} (h_u(\mathbf{u}_t, \mathbf{v}_t))_{\mathcal{I}} + c(\mathbf{A}\mathbf{u}_t)_{\mathcal{I}} \\ (h_v(\mathbf{u}_t, \mathbf{v}_t))_{\mathcal{I}} + (\mathbf{A}\mathbf{v}_t)_{\mathcal{I}} \end{pmatrix}, \right. \\ &\quad \left. \delta t \begin{pmatrix} \sigma_u^2 \mathbf{I}_{|\mathcal{I}|} & \mathbf{0} \\ \mathbf{0} & \sigma_v^2 \mathbf{I}_{|\mathcal{I}|} \end{pmatrix}\right), \end{aligned}$$

where $\mathbf{A}_{\mathcal{I}}$ denotes the submatrix of $\mathbf{A}$ obtained by restricting its rows to the index set $\mathcal{I}$. We choose the time step δt of the Euler-Maruyama method to be exactly the time step used in the partial evolution scheme.

Define the one-hop neighborhood of $\mathcal{I}$ by $\mathcal{I}^+ := (\mathcal{I} \cup \{k-1, l+1\}) \cap \{1, \dots, 200\}$. For comparison, the partial evolution scheme updates only the variables within the patch

$\mathcal{I}$ using information from its local neighborhood $\mathcal{I}^+$,

$$\begin{aligned} \mathbf{u}_{t+\delta t, \mathcal{I}} &= \mathbf{u}_{t, \mathcal{I}} + \delta t(h_u(\mathbf{u}_{t, \mathcal{I}}, \mathbf{v}_{t, \mathcal{I}}) + c\mathbf{A}_{\mathcal{I}, \mathcal{I}^+} \mathbf{u}_{t, \mathcal{I}^+}) + \sigma_u \delta \mathbf{B}_{t, \mathcal{I}}^u, \\ \mathbf{v}_{t+\delta t, \mathcal{I}} &= \mathbf{v}_{t, \mathcal{I}} + \delta t(h_v(\mathbf{u}_{t, \mathcal{I}}, \mathbf{v}_{t, \mathcal{I}}) + \mathbf{A}_{\mathcal{I}, \mathcal{I}^+} \mathbf{v}_{t, \mathcal{I}^+}) + \sigma_v \delta \mathbf{B}_{t, \mathcal{I}}^v. \end{aligned}$$

where $\mathbf{u}_{t, \mathcal{I}^+}$ denotes the restriction of $\mathbf{u}_t$ to the index set $\mathcal{I}^+$, and $\mathbf{A}_{\mathcal{I}, \mathcal{I}^+}$ is the submatrix of $\mathbf{A}$ obtained by restricting its rows to the index set $\mathcal{I}$ and its columns to $\mathcal{I}^+$. Therefore, the conditional distribution induced by the partial evolution scheme is

$$\begin{aligned} q(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t) &= \mathcal{N}(\boldsymbol{\mu}_{t, \mathcal{I}}, \boldsymbol{\Sigma}_{\mathcal{I}}), \\ \boldsymbol{\mu}_{t, \mathcal{I}} &= \begin{pmatrix} \mathbf{u}_{t, \mathcal{I}} \\ \mathbf{v}_{t, \mathcal{I}} \end{pmatrix} \\ &\quad + \delta t \begin{pmatrix} h_u(\mathbf{u}_{t, \mathcal{I}}, \mathbf{v}_{t, \mathcal{I}}) + c\mathbf{A}_{\mathcal{I}, \mathcal{I}^+} \mathbf{u}_{t, \mathcal{I}^+} \\ h_v(\mathbf{u}_{t, \mathcal{I}}, \mathbf{v}_{t, \mathcal{I}}) + \mathbf{A}_{\mathcal{I}, \mathcal{I}^+} \mathbf{v}_{t, \mathcal{I}^+} \end{pmatrix}, \\ \boldsymbol{\Sigma}_{\mathcal{I}} &= \delta t \begin{pmatrix} \sigma_u^2 \mathbf{I}_{|\mathcal{I}|} & \mathbf{0} \\ \mathbf{0} & \sigma_v^2 \mathbf{I}_{|\mathcal{I}|} \end{pmatrix}. \end{aligned}$$

Comparing $\hat{q}(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t)$ and $q(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t)$, we note that

$$\begin{aligned} h_u(\mathbf{u}_{t, \mathcal{I}}, \mathbf{v}_{t, \mathcal{I}}) &= (h_u(\mathbf{u}_t, \mathbf{v}_t))_{\mathcal{I}}, \\ h_v(\mathbf{u}_{t, \mathcal{I}}, \mathbf{v}_{t, \mathcal{I}}) &= (h_v(\mathbf{u}_t, \mathbf{v}_t))_{\mathcal{I}}, \end{aligned}$$

since h_u and h_v are applied pointwise. Moreover, the equality

$$(\mathbf{A}\mathbf{u}_t)_{\mathcal{I}} = \mathbf{A}_{\mathcal{I}, \mathcal{I}^+} \mathbf{u}_{t, \mathcal{I}^+}$$

holds because the discrete Laplacian matrix $\mathbf{A}$ is tridiagonal under the Neumann boundary discretization, so that each component depends only on its immediate neighbors.

Together, these observations imply that the conditional distribution induced by the partial evolution scheme matches the marginal of the full evolution on the patch $\mathcal{I}$, i.e.,

$$q(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t) = \hat{q}(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t).$$

Step 2: Conditional independence of $\Delta \mathbf{z}_{t, \mathcal{I}}$ and $\Delta \mathbf{z}_{t, \mathcal{J}}$ when $\mathcal{I} \neq \mathcal{J}$.

From the definition of $\hat{q}(\mathbf{x}_{t+\delta t, \mathcal{I}} | \mathbf{x}_t)$, we can see that $\mathbf{x}_{t+\delta t, \mathcal{I}} - \mathbf{x}_t$ is independent of $\mathbf{x}_{t+\delta t, \mathcal{J}} - \mathbf{x}_t$ when $\mathcal{I} \neq \mathcal{J}$. Therefore, $\Delta \mathbf{z}_{t, \mathcal{I}} = \boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{I}}) - \boldsymbol{\varphi}(\mathbf{x}_t, \mathcal{I})$ is conditionally independent of $\Delta \mathbf{z}_{t, \mathcal{J}} = \boldsymbol{\varphi}(\mathbf{x}_{t+\delta t, \mathcal{J}}) - \boldsymbol{\varphi}(\mathbf{x}_t, \mathcal{J})$.

Step 3: $\Delta \mathbf{z}_t^* \approx \frac{1}{K} \sum_{\mathcal{I}} \Delta \mathbf{z}_{t, \mathcal{I}}^*$

The macroscopic observables $\mathbf{z}^*$ are chosen to be $(\frac{1}{200} \sum_{i=1}^{200} u(x_i, t), \frac{1}{200} \sum_{i=1}^{200} v(x_i, t))$. By definition, then the global average equals the average of the patch averages. ■

- [1] J. Ordonez-Miranda, Y. Ezzahri, K. Joulain, J. Drevillon, and J. Alvarado-Gil, Modeling of the electrical conductivity, thermal conductivity, and specific heat capacity of VO₂, *Phys. Rev. B* **98**, 075144 (2018).
- [2] J. D. Durrant and J. A. McCammon, Molecular dynamics simulations and drug discovery, *BMC Biol.* **9**, 71 (2011).
- [3] S. A. Hollingsworth and R. O. Dror, Molecular dynamics simulation for all, *Neuron* **99**, 1129 (2018).
- [4] M. O. Steinhauser and S. Hiermaier, A review of computational methods in materials science: Examples from shock-wave and polymer physics, *Int. J. Mol. Sci.* **10**, 5135 (2009).

- [5] D. K. Watson and M. Dunn, Rearranging the exponential wall for large N -body systems, *Phys. Rev. Lett.* **105**, 020402 (2010).
- [6] A. J. Cohen, P. Mori-Sánchez, and W. Yang, Challenges for density functional theory, *Chem. Rev.* **112**, 289 (2012).
- [7] M. Stamatakis and D. G. Vlachos, Unraveling the complexity of catalytic reactions via kinetic Monte Carlo simulation: Current status and frontiers, *ACS Catal.* **2**, 2648 (2012).
- [8] M. Andersen, C. Panosetti, and K. Reuter, A practical guide to surface kinetic Monte Carlo simulations, *Front. Chem.* **7**, 202 (2019).

- [9] M. Pineda and M. Stamatakis, Kinetic Monte Carlo simulations for heterogeneous catalysis: Fundamentals, current status, and challenges, *J. Chem. Phys.* **156**, 120902 (2022).
- [10] K. Li, H. Shang, Y. Zhang, S. Li, and B. Wu, OpenKMC: A KMC design for hundred-billion-atom simulation using millions of cores on Sunway TaihuLight, in *SC '19: The International Conference for High Performance Computing, Networking, Storage, and Analysis* (ACM, New York, NY, 2019), pp. 1–16.
- [11] H. Shang, X. Chen, X. Gao, R. Lin, and L. Wang, TensorKMC: Kinetic Monte Carlo simulation of 50 trillion atoms driven by deep learning on a new generation of Sunway supercomputer, in *SC '21: The International Conference for High Performance Computing, Networking, Storage and Analysis* (ACM, New York, NY, 2021), pp. 1–14.
- [12] 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).
- [13] S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt, and B. Kozinsky, E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials, *Nat. Commun.* **13**, 2453 (2022).
- [14] J. Behler and M. Parrinello, Generalized neural-network representation of high-dimensional potential-energy surfaces, *Phys. Rev. Lett.* **98**, 146401 (2007).
- [15] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, Neural message passing for quantum chemistry, in *International Conference on Machine Learning* (PMLR, Cambridge, MA, 2017), pp. 1263–1272.
- [16] W. Jia, H. Wang, M. Chen, D. Lu, L. Lin, R. Car, W. E, and L. Zhang, Pushing the limit of molecular dynamics with ab initio accuracy to 100 million atoms with machine learning, in *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis* (IEEE, Piscataway, NJ, 2020), pp. 1–14.
- [17] A. E. Durumeric, N. E. Charron, C. Templeton, F. Musil, and K. Bonneau, Machine learned coarse-grained protein force-fields: Are we there yet? *Curr. Opin. Struct. Biol.* **79**, 102533 (2023).
- [18] B. E. Husic, N. E. Charron, D. Lemm, J. Wang, and A. Pérez, Coarse graining molecular dynamics with graph neural networks, *J. Chem. Phys.* **153**, 194101 (2020).
- [19] J. Köhler, Y. Chen, A. Krämer, C. Clementi, and F. Noé, Flow-matching: Efficient coarse-graining of molecular dynamics without forces, *J. Chem. Theory Comput.* **19**, 942 (2023).
- [20] J. Wang, S. Olsson, C. Wehmeyer, A. Pérez, and N. E. Charron, Machine learning of coarse-grained molecular dynamics force fields, *ACS Cent. Sci.* **5**, 755 (2019).
- [21] L. Zhang, J. Han, H. Wang, R. Car, and W. E., DeepCG: Constructing coarse-grained models via deep neural networks, *J. Chem. Phys.* **149**, 034101 (2018).
- [22] X. Chen, B. W. Soh, Z.-E. Ooi, E. Vissol-Gaudin, H. Yu, K. S. Novoselov, K. Hippalgaonkar, and Q. Li, Constructing custom thermodynamics using deep learning, *Nat. Comput. Sci.* **4**, 66 (2023).
- [23] M. Chen and Q. Li, Learning macroscopic dynamics from partial microscopic observations, *Adv. Neural Inf. Process. Syst.* **37**, 48996 (2024).
- [24] N. G. Van Kampen, *Stochastic Processes in Physics and Chemistry* (Elsevier, Amsterdam, 1992), Vol. 1.
- [25] X. Fu, T. Xie, N. J. Rebello, B. Olsen, and T. S. Jaakkola, Simulate time-integrated coarse-grained molecular dynamics with multi-scale graph networks, *Transact. Mach. Learn. Res.* (2023).
- [26] A. Boral, Z. Y. Wan, L. Zepeda-Núñez, J. Lottes, Q. Wang, Y.-f. Chen, J. Anderson, and F. Sha, Neural ideal large eddy simulation: Modeling turbulence with neural stochastic differential equations, *Adv. Neural Inf. Process. Syst.* **36**, 69270 (2023).
- [27] J. Fish, *Multiscale Methods: Bridging the Scales in Science and Engineering* (Oxford University Press, Oxford, 2009).
- [28] C. W. Gear, J. M. Hyman, P. G. Kevrekidis, I. G. Kevrekidis, O. Runborg, and C. Theodoropoulos, Equation-free, coarse-grained multiscale computation: Enabling microscopic simulators to perform system-level analysis, *Commun. Math. Sci.* **1**, 715 (2003).
- [29] P. Liu, G. Samaey, C. W. Gear, and I. G. Kevrekidis, On the acceleration of spatially distributed agent-based computations: A patch dynamics scheme, *Appl. Numer. Math.* **92**, 54 (2015).
- [30] G. Samaey, I. G. Kevrekidis, and D. Roose, *Damping Factors for the Gap-Tooth Scheme*, in *Multiscale Modelling and Simulation* (Springer, Berlin, Heidelberg, 2004), pp. 93–102.
- [31] G. Samaey, D. Roose, and I. G. Kevrekidis, The gap-tooth scheme for homogenization problems, *Multiscale Model. Simul.* **4**, 278 (2005).
- [32] G. Samaey, I. G. Kevrekidis, and D. Roose, Patch dynamics with buffers for homogenization problems, *J. Comput. Phys.* **213**, 264 (2006).
- [33] H. Yu, X. Tian, W. E, and Q. Li, OnsagerNet: Learning stable and interpretable dynamics using a generalized Onsager principle, *Phys. Rev. Fluids* **6**, 114402 (2021).
- [34] F. Dietrich, A. Makeev, G. Kevrekidis, N. Evangelou, and T. Bertalan, Learning effective stochastic differential equations from microscopic simulations: Linking stochastic numerics to deep learning, *Chaos* **33**, 023121 (2023).
- [35] Z. Zhang, Y. Shin, and G. Em Karniadakis, GFNNs: Generic formalism informed neural networks for deterministic and stochastic dynamical systems, *Philos. Trans. R. Soc. A* **380**, 20210207 (2022).
- [36] T.-T. Gao, B. Barzel, and G. Yan, Learning interpretable dynamics of stochastic complex systems from experimental data, *Nat. Commun.* **15**, 6029 (2024).
- [37] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, A kernel two-sample test, *J. Mach. Learn. Res.* **13**, 723 (2012).
- [38] P. Kidger, J. Foster, X. Li, and T. J. Lyons, Neural SDEs as infinite-dimensional GANs, in *International Conference on Machine Learning* (PMLR, Cambridge, MA, 2021), pp. 5453–5463.
- [39] S. G. Brush, History of the Lenz-Ising model, *Rev. Mod. Phys.* **39**, 883 (1967).
- [40] B. A. Cipra, An introduction to the Ising model, *Am. Math. Mon.* **94**, 937 (1987).
- [41] A. W. Sandvik, Computational studies of quantum spin systems, *AIP Conf. Proc.* **1297**, 135 (2010).
- [42] B. Xing, T. J. Rupert, X. Pan, and P. Cao, Neural network kinetics for exploring diffusion multiplicity and chemical ordering in compositionally complex materials, *Nat. Commun.* **15**, 3879 (2024).

- [43] J. Cowley, Short-range order and long-range order parameters, *Phys. Rev.* **138**, A1384 (1965).
- [44] A. Fernández-Caballero, J. Wróbel, P. Mummery, and D. Nguyen-Manh, Short-range order in high entropy alloys: Theoretical formulation and application to Mo-Nb-Ta-V-W system, *J. Phase Equilib. Diffus.* **38**, 391 (2017).
- [45] Y. Han, H. Chen, Y. Sun, J. Liu, S. Wei, B. Xie, Z. Zhang, Y. Zhu, M. Li, J. Yang, *et al.*, Ubiquitous short-range order in multi-principal element alloys, *Nat. Commun.* **15**, 6486 (2024).
- [46] J. Crank, *The Mathematics of Diffusion* (Oxford University Press, Oxford, 1979).
- [47] S. R. De Groot and P. Mazur, *Non-Equilibrium Thermodynamics* (Dover Publications, New York, 1984).
- [48] J. W. Cahn and J. E. Hilliard, Free energy of a nonuniform system. I. Interfacial free energy, *J. Chem. Phys.* **28**, 258 (1958).
- [49] S. M. Allen and J. W. Cahn, A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening, *Acta Metall.* **27**, 1085 (1979).
- [50] L.-Q. Chen, Phase-field models for microstructure evolution, *Annu. Rev. Mater. Res.* **32**, 113 (2002).
- [51] M. Tsige and G. S. Grest, Molecular dynamics simulation of solvent-polymer interdiffusion: Fickian diffusion, *J. Chem. Phys.* **120**, 2989 (2004).
- [52] X. Liu, A. Martín-Calvo, E. McGarrity, S. K. Schnell, and S. Calero, Fick diffusion coefficients in ternary liquid systems from equilibrium molecular dynamics simulations, *Ind. Eng. Chem. Res.* **51**, 10247 (2012).
- [53] W.-M. Choi, Y. H. Jo, S. S. Sohn, S. Lee, and B.-J. Lee, Understanding the physical metallurgy of the CoCrFeMnNi high-entropy alloy: An atomistic simulation study, *npj Comput. Mater.* **4**, 1 (2018).
- [54] T. Kostiuhenko, F. Körmann, J. Neugebauer, and A. Shapeev, Impact of lattice relaxations on phase transitions in a high-entropy alloy studied by machine-learning potentials, *npj Comput. Mater.* **5**, 55 (2019).
- [55] Y. Cao, K. Sheriff, and R. Freitas, Capturing short-range order in high-entropy alloys with machine learning potentials, *npj Comput. Mater.* **11**, 268 (2025).
- [56] K. Kremer and K. Binder, Monte Carlo simulation of lattice models for macromolecules, *Comput. Phys. Rep.* **7**, 259 (1988).
- [57] T. E. Gartner III and A. Jayaraman, Modeling and simulations of polymers: A roadmap, *Macromolecules* **52**, 755 (2019).
- [58] <https://github.com/mengyi-chen/Scalable-learning-of-macroscopic-stochastic-dynamics.git>.
- [59] M. Kochmański, T. Paszkiewicz, and S. Wolski, Curie-Weiss magnet—A simple model of phase transition, *Eur. J. Phys.* **34**, 1555 (2013).
- [60] R. S. Ellis, *Entropy, Large Deviations, and Statistical Mechanics* (Springer Science & Business Media, New York, NY, 2012), Vol. 271.