Learning moment closure in reaction-diffusion systems with spatial dynamic Boltzmann distributions

Oliver K. Ernst
Department of Physics, University of California at San Diego, La Jolla, California 92093, USA

Thomas M. Bartol
Salk Institute for Biological Studies, La Jolla, California 92037, USA

Terrence J. Sejnowski
Salk Institute for Biological Studies, La Jolla, California 92037, USA
and Division of Biological Sciences, University of California at San Diego, La Jolla, California 92093, USA

Eric Mjolsness
Departments of Computer Science and Mathematics, and Institute for Genomics and Bioinformatics, University of California at Irvine, Irvine, 92697 California, USA

(Received 22 April 2019; published 26 June 2019)

Many physical systems are described by probability distributions that evolve in both time and space. Modeling these systems is often challenging due to their large state space and analytically intractable or computationally expensive dynamics. To address these problems, we study a machine-learning approach to model reduction based on the Boltzmann machine. Given the form of the reduced model Boltzmann distribution, we introduce an autonomous differential equation system for the interactions appearing in the energy function. The reduced model can treat systems in continuous space (described by continuous random variables), for which we formulate a variational learning problem using the adjoint method to determine the right-hand sides of the differential equations. This approach can be used to enforce a reduced physical model by a suitable parametrization of the differential equations. The parametrization we employ uses the basis functions from finite-element methods, which can be used to model any physical system. One application domain for such physics-informed learning algorithms is to modeling reaction-diffusion systems. We study a lattice version of the Rössler chaotic oscillator, which illustrates the accuracy of the moment closure approximation made by the method and its dimensionality reduction power.

DOI: 10.1103/PhysRevE.99.063315

I. INTRODUCTION

Probability distributions that evolve in both space and time appear in many modeling applications, such as reaction-diffusion systems [1–4], neural population activities [5,6], and fluid dynamics [7], as well as in engineering fields such as traffic forecasting [8] and navigation of autonomous vehicles [9]. However, (1) the state space of such distributions is generally large, and (2) the dynamical systems obeyed by their observables may be unknown or intractable to solve analytically. These aspects make modeling spatiotemporal systems a computational challenge and limit the interpretability of such models.

Reaction-diffusion systems are a typical example of these problems. The distribution over system states obeys a chemical master equation (CME) [10], but the state space grows exponentially with the number of random variables that describe it [11]. Further, the time evolution of observables is not closed, i.e., the time evolution of lower-order moments depends on higher-order ones (similar to a BBGKY hierarchy [12]). Their estimation therefore requires the use of a moment closure approximation (e.g., Refs. [13,14] and others; see Ref. [15] for a review), or otherwise sampling algorithms such as the Gillespie stochastic simulation algorithm (SSA) [16] or related methods for spatial systems [17,18].

A reduced model is one which approximates both the true distribution and its dynamics and should address the challenges above by (1) having a smaller state space and (2) being more easily tractable or computationally efficient [15]. Reduced models of reaction-diffusion systems are widely studied [1,19], particularly in multiscale modeling in biology [20]. Recent work [2,4,13] has demonstrated methods based on entropic matching as a highly general approach to model reduction of reaction networks.

In this paper, we demonstrate a machine-learning (ML) approach to model reduction using Boltzmann machines (BMs) [21]. We formalize the methods of earlier work [15,22] and extend these with the introduction of latent variables. Our approach also extends work on entropic matching methods to treat spatial systems. We present examples for spatial chemical reaction systems that demonstrate the moment closure properties of the reduced model and apply the method to learn a spatial chaotic oscillator.

The area of ML most suited for model reduction of reaction-diffusion systems are generative models [23], where it is assumed that data are samples of an unknown probability distribution, with the goal of estimating this distribution by a structured approach. This structure can offer insight into the problem that has not been obtainable analytically [24] and allows new samples to be drawn using, e.g., Markov-chain Monte Carlo methods [25]. Typically, a graphical model for the distribution is introduced and learned by determining interaction parameters between random variables. Similar ML approaches have emerged as a powerful tool for studying quantum many-body problems [26,27].

Our approach introduces a differential equation (DE) model for interaction parameters in the graph. The learning problem is formulated to determine these DEs by a maximum likelihood approach. In contrast to ML methods for learning temporal data such as recurrent networks, here prior information about the system may be used to enforce a reduced physical model by parametrizing the functional forms of the DEs.

A further advantage of this strategy is that it offers a natural description of systems where neither time nor space are discretized, i.e., the system is described by random variables representing space continuously and varying continuously in time. In this case, a partial differential equation (PDE) model can be introduced. Spatially continuous descriptions are beneficial when confined geometries would introduce error into lattice-based methods, e.g., when modeling reaction-diffusion systems at synapses [17].

The algorithmic solution to this learning problem takes the form of a PDE-constrained optimization problem. The algorithm and its derivation are closely related to BM learning, but in this case data samples are trajectories in space and time rather than instantaneous snapshots or slices. A related framework, graph-constrained correlation dynamics [15], has a similar learning goal but uses spatially aggregated snapshots in time and does not consider spatial reduced models.

The outline of this paper is as follows: (1) in Sec. II we introduce spatial dynamic Boltzmann distributions as reduced models of reaction-diffusion systems in continuous space and formulate their learning problem using the adjoint method; (2) in Sec. III we demonstrate the connection to a restricted Boltzmann machine; (3) in Sec. IV we show how hidden layers implement moment closure approximations and apply the method to a spatial chaotic oscillator.

II. SPATIAL DYNAMIC BOLTZMANN DISTRIBUTIONS

In this section, we introduce the reduced model for a spatiotemporal distribution and its dynamics in continuous space from Ref. [22] and formulate the learning problem using the adjoint method. We consider the specific application of a reaction-diffusion system but note that the methods are also applicable to other spatiotemporal systems.

The state of a reaction-diffusion system at some time t is described by n particles of species labels $\boldsymbol{\alpha}$ located at positions $\boldsymbol{x}$ in generally continuous three-dimensional (3D) space (each x_i for $i = 1, \ldots, n$ is a coordinate in 3D space). Let the true distribution over system states be denoted by $p(n, \boldsymbol{\alpha}, \boldsymbol{x}, t)$, whose time evolution can be described using the Doi-Peliti formalism [46].

To define the reduced model, introduce k-particle interaction functions $\nu_k(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t)$, where $\langle i\rangle_k^n$ denotes any ordered subset of k indexes with each index in $\{1, \ldots, n\}$. Given a set of such interaction functions $\{\nu\}_{k=1}^K$ up to cutoff order K, define a *spatial dynamic Boltzmann distribution* as one of the form:

$$\tilde{p}(n, \boldsymbol{\alpha}, \boldsymbol{x}, t; \{\nu\}) = \frac{1}{Z[\{\nu\}]} \exp\left[-\sum_{k=1}^{K} \sum_{\langle i\rangle_k^n} \nu_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right) \right], \tag{1}$$

where the sum over $\langle i\rangle_k^n$ iterates over unique kth-order interactions between n particles, and the partition function is

$$Z[\{\nu\}] = \sum_{n=0}^{\infty} \sum_{\boldsymbol{\alpha}} \int d\boldsymbol{x}\ \exp\left[-\sum_{k=1}^{K} \sum_{\langle i\rangle_k^n} \nu_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right) \right]. \tag{2}$$

Boltzmann distributions are maximum entropy (MaxEnt) distributions, where each interaction function $\nu_k(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t)$ controls a corresponding moment $\mu_k(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t)$, given by:

$$\mu_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right) = \sum_{n'=0}^{\infty} \sum_{\boldsymbol{\alpha}'} \int d\boldsymbol{x}'\ p(n', \boldsymbol{\alpha}', \boldsymbol{x}', t) \times \sum_{\langle j\rangle_k^{n'}} \delta\left(\boldsymbol{x}_{\langle i\rangle_k^n} - \boldsymbol{x}'_{\langle j\rangle_k^{n'}}\right) \delta\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n} - \boldsymbol{\alpha}'_{\langle j\rangle_k^{n'}}\right), \tag{3}$$

that is, the average number of k-sized tuplets of particles of species $\boldsymbol{\alpha}_{\langle i\rangle_k^n}$ at locations $\boldsymbol{x}_{\langle i\rangle_k^n}$. Note that $\boldsymbol{\alpha}'$ and $\boldsymbol{x}'$ are of size n'.

A. Moment matching

Given a set of training data drawn from $p(n, \boldsymbol{\alpha}, \boldsymbol{x}, t)$ at some instant in time, the BM learning algorithm determines parameters in the energy function such that the instantaneous distribution (1) is the MaxEnt distribution consistent with the moments in the data set. To learn a reduced model of a system that evolves in both time and space continuously, we seek the distribution that is *at all times* the MaxEnt solution. Define as the action the Kullback-Leibler (KL) divergence $\mathcal{D}_{\mathrm{KL}}$ between the true and reduced models, p and $\tilde{p}$, integrated over all times:

$$S = \int_{t_0}^{t_f} dt\ \mathcal{D}_{\mathrm{KL}}(p||\tilde{p}), \tag{4}$$

where the Lagrangian is $\mathcal{L}(t; \{\nu\}) = \mathcal{D}_{\mathrm{KL}}(p||\tilde{p})$ for

$$\mathcal{D}_{\mathrm{KL}}(p||\tilde{p}) = \sum_{n=0}^{\infty} \sum_{\boldsymbol{\alpha}} \int d\boldsymbol{x} \times p(n, \boldsymbol{\alpha}, \boldsymbol{x}, t) \ln \frac{p(n, \boldsymbol{\alpha}, \boldsymbol{x}, t)}{\tilde{p}(n, \boldsymbol{\alpha}, \boldsymbol{x}, t; \{\nu\})}. \tag{5}$$

Minimizing S is thus equivalent to maximizing the integrated log-likelihood of the observed data given the interaction functions. Other approaches for modeling time series are discussed in Sec. III A.

The condition for extremizing the action follows from the chain rule as

$$\delta S = \int_{t_0}^{t_f} dt \sum_{n=0}^{\infty} \sum_{\boldsymbol{\alpha}} \int d\boldsymbol{x} \times \sum_{k=1}^{K} \sum_{\langle i\rangle_k^n} \Delta\mu_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right) \delta\nu_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right) = 0, \tag{6}$$

where

$$\Delta\mu_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right) = \tilde{\mu}_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right) - \mu_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right), \tag{7}$$

where μ and $\tilde{\mu}$ are averages taken over p and $\tilde{p}$. This appearance of a difference of moments is the common result from using the KL divergence in the objective functional.

B. An adjoint method learning problem for spatial dynamic Boltzmann distributions

Introduce for each interaction function $\nu_k(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t)$ a *functional* model:

$$\frac{d}{dt}\nu_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right) = \mathcal{F}_k[\{\nu\}](\boldsymbol{\alpha}, \boldsymbol{x}, t), \tag{8}$$

with initial condition $\nu_k(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t_0) = \eta_k(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n})$ and where $\{\nu\} = \{\nu_k\}_{k=1}^K$ denotes possibly all interaction functions. We use $\mathcal{F}$ to denote a functional, allowing, for example, a PDE model to be introduced. Note that the arguments to the left-hand side may also appear on the right, for example, through a spatial derivative term $\nabla\nu_k(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t)$.

Introduce vector notation[1] $\boldsymbol{\nu}(\boldsymbol{\alpha}, \boldsymbol{x}, t)$ and $\boldsymbol{\mathcal{F}}[\{\nu\}](\boldsymbol{\alpha}, \boldsymbol{x}, t)$ for the left- and right-hand sides of (8), which contain $N = \sum_{k=1}^K \binom{n}{k}$ entries, one for every possible $(k, \langle i\rangle_k^n)$ in some order $i = 1, \ldots, N$. To enforce the constraint (8), define the Lagrangian as the functional:

$$\mathcal{L}[\{\nu\}, \{\zeta\}](t) = \mathcal{D}_{\mathrm{KL}}(p||\tilde{p}) + \sum_{n=0}^{\infty} \sum_{\boldsymbol{\alpha}} \int d\boldsymbol{x}\, \boldsymbol{\zeta}^{\intercal}(\boldsymbol{\alpha}, \boldsymbol{x}, t) \times \left[\frac{d\boldsymbol{\nu}(\boldsymbol{\alpha}, \boldsymbol{x}, t)}{dt} - \boldsymbol{\mathcal{F}}[\{\nu\}](\boldsymbol{\alpha}, \boldsymbol{x}, t)\right], \tag{9}$$

where we have introduced Lagrange multiplier functions $\boldsymbol{\zeta}(\boldsymbol{\alpha}, \boldsymbol{x}, t)$ corresponding to $\boldsymbol{\nu}(\boldsymbol{\alpha}, \boldsymbol{x}, t)$ and $\{\zeta\} = \{\zeta_k\}_{k=1}^K$. Since the constraint is satisfied, then the action is as before $S = \int_{t_0}^{t_f} dt\ \mathcal{L}[\{\nu\}, \{\zeta\}](t)$.

Introducing perturbations $\delta\boldsymbol{\nu}(\boldsymbol{\alpha}, \boldsymbol{x}, t)$ to the interaction functions gives as condition for extremizing the action:

$$\delta S = \int_{t_0}^{t_f} dt \sum_{n=0}^{\infty} \sum_{\boldsymbol{\alpha}} \int d\boldsymbol{x}\, \delta\boldsymbol{\nu}^{\intercal}(\boldsymbol{\alpha}, \boldsymbol{x}, t) \times \left\{\Delta\boldsymbol{\mu}(\boldsymbol{\alpha}, \boldsymbol{x}, t) - \frac{d\boldsymbol{\zeta}(\boldsymbol{\alpha}, \boldsymbol{x}, t)}{dt} - \frac{\delta\mathcal{J}[\{\nu\}, \{\zeta\}](t)}{\delta\boldsymbol{\nu}(\boldsymbol{\alpha}, \boldsymbol{x}, t)}\right\} = 0, \tag{10}$$

where the boundary terms from the integration by parts in the second term have vanished due to the boundary condition for the adjoint variables $\boldsymbol{\zeta}(\boldsymbol{\alpha}, \boldsymbol{x}, t_f) = 0$, and we have defined:

$$\mathcal{J}[\{\nu\}, \{\zeta\}](t) = \sum_{n'=0}^{\infty} \sum_{\boldsymbol{\alpha}'} \int d\boldsymbol{x}' \times \boldsymbol{\zeta}^{\intercal}(\boldsymbol{\alpha}', \boldsymbol{x}', t)\boldsymbol{\mathcal{F}}[\{\nu\}](\boldsymbol{\alpha}', \boldsymbol{x}', t). \tag{11}$$

From (10) we obtain the adjoint system

$$\frac{d\boldsymbol{\zeta}(\boldsymbol{\alpha}, \boldsymbol{x}, t)}{dt} = \Delta\boldsymbol{\mu}(\boldsymbol{\alpha}, \boldsymbol{x}, t) - \frac{\delta\mathcal{J}[\{\nu\}, \{\zeta\}](t)}{\delta\boldsymbol{\nu}(\boldsymbol{\alpha}, \boldsymbol{x}, t)}. \tag{12}$$

Depending on the form of the functional, additional boundary conditions may be enforced to evaluate the term on the right. Equations (8) and (12) can be equivalently expressed by the Hamiltonian system

$$\frac{d\boldsymbol{\nu}(\boldsymbol{\alpha}, \boldsymbol{x}, t)}{dt} = \frac{\delta H[\{\nu\}, \{\zeta\}](t)}{\delta\boldsymbol{\zeta}(\boldsymbol{\alpha}, \boldsymbol{x}, t)}, \quad \frac{d\boldsymbol{\zeta}(\boldsymbol{\alpha}, \boldsymbol{x}, t)}{dt} = -\frac{\delta H[\{\nu\}, \{\zeta\}](t)}{\delta\boldsymbol{\nu}(\boldsymbol{\alpha}, \boldsymbol{x}, t)}, \tag{13}$$

where

$$H[\{\nu\}, \{\zeta\}](t) = -\mathcal{D}_{\mathrm{KL}}(p||\tilde{p}) + \mathcal{J}[\{\nu\}, \{\zeta\}](t). \tag{14}$$

Given a reduced model for the dynamics (8), Eq. (10) gives the necessary condition for extremizing the action. In a typical model reduction setting, however, the reduced model is not known beforehand. What should the form of the model (8) be to extremize the action (4)? Consider the case where the functional is specified in terms of some ordinary functions. We next set up a variational problem for these functions appearing on the right-hand side of the differential equation. Variational problems of this form have been studied previously, first in the context of optimal control theory [28,29] and later didactically in Ref. [30].

Let the functional be of the form:

$$\frac{d}{dt}\nu_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right) = \mathcal{F}_k[\{\nu\}, \{F_k\}](\boldsymbol{\alpha}, \boldsymbol{x}, t), \tag{15}$$

where the M_k ordinary functions appearing on the right-hand side are $F_k^{(s)}(\{\nu(\boldsymbol{\alpha}, \boldsymbol{x}, t)\})$ for $s = 1, \ldots, M_k$, denoted by $\{F_k\} = \{F_k^{(s)}\}_{s=1}^{M_k}$. For arbitrary perturbations $\delta F_k^{(s)}$, extremizing the action gives

$$\delta S = -\int_{t_0}^{t_f} dt \sum_{n=0}^{\infty} \sum_{\boldsymbol{\alpha}} \int d\boldsymbol{x} \sum_{k=1}^{K} \sum_{\langle i\rangle_k^n} \sum_{s=1}^{M_k} \frac{\delta\mathcal{J}[\{\nu\}, \{\zeta\}](t)}{\delta F_k^{(s)}(\{\nu(\boldsymbol{\alpha}, \boldsymbol{x}, t)\})} \times \delta F_k^{(s)}(\{\nu(\boldsymbol{\alpha}, \boldsymbol{x}, t)\}) = 0. \tag{16}$$

Equation (16) is the variational calculus form of the sensitivity equation obtained by the adjoint method when the functional model is specified in terms of some parameter vector [31]. This is particularly clear if we consider the specific form of (15) as the autonomous ordinary differential equation (ODE) system:

$$\frac{d}{dt}\nu_k\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right) = F_k\left(\left\{\nu\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right)\right\}\right), \tag{17}$$

[1] In this notation, the dot product is $\boldsymbol{a}^{\intercal}(\boldsymbol{\alpha}, \boldsymbol{x})\boldsymbol{b}(\boldsymbol{\alpha}, \boldsymbol{x}) = \sum_{k=1}^K \sum_{\langle i\rangle_k^n} a(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}) b(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n})$.

where $\{\nu(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t)\}$ denotes all ν of all possible arguments appearing on the left-hand side. In this case, (16) becomes

$$\delta S = -\int_{t_0}^{t_f} dt \sum_{n=0}^{\infty} \sum_{\boldsymbol{\alpha}} \int d\boldsymbol{x} \times [\boldsymbol{\zeta}^{\intercal}(\boldsymbol{\alpha}, \boldsymbol{x}, t)\delta \boldsymbol{F}(\{\nu(\boldsymbol{\alpha}, \boldsymbol{x}, t)\})] = 0, \tag{18}$$

where as before we have used vectors of length N to denote possible $(k, \langle i\rangle_k^n)$ as before. This resembles the adjoint method sensitivity equation, where variational terms δF_k and δS replace ordinary derivatives with respect to parameters. This will be pursued further in Sec. III A. From (18) follows the common result that extremizing the action requires that the adjoint variables vanish everywhere $\zeta_k(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t) = 0$. One case when this is satisfied is if the adjoint system is source free $\Delta\mu_k(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t) = 0$, i.e., the moment matching condition is met.

From the Euler-Lagrange equations (12), the adjoint variables obey:

$$\frac{d\boldsymbol{\zeta}(\boldsymbol{\alpha}, \boldsymbol{x}, t)}{dt} = \Delta\boldsymbol{\mu}(\boldsymbol{\alpha}, \boldsymbol{x}, t) - G^{\intercal}(\boldsymbol{\alpha}, \boldsymbol{x}, t)\boldsymbol{\zeta}(\boldsymbol{\alpha}, \boldsymbol{x}, t), \tag{19}$$

where the elements of the $N \times N$ matrix G are

$$G_{i,i'}(\boldsymbol{\alpha}, \boldsymbol{x}, t) = \frac{\partial F_k\left[\left\{\nu\left(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t\right)\right\}\right]}{\partial \nu_{k'}\left(\boldsymbol{\alpha}_{\langle i\rangle_{k'}^n}, \boldsymbol{x}_{\langle i\rangle_{k'}^n}, t\right)}, \tag{20}$$

where $(k, \langle i\rangle_k^n)$ corresponds to index i and $(k', \langle i\rangle_{k'}^n)$ corresponds to index i'. Appendix A gives the formal solution to (19) and makes explicit the connection between the conditions for extrema (18) and (6).

III. DYNAMICS FOR RESTRICTED BOLTZMANN MACHINES

We next consider a specific case of the formalism of Sec. II where the system is described by discrete random variables. A Boltzmann distribution on a state $\boldsymbol{v} = \{v_1, \ldots, v_N\}$ of N discrete random variables is of the form:

$$\tilde{p}(\boldsymbol{v}) = \frac{1}{Z}\exp[-E(\boldsymbol{v})], \tag{21}$$

where Z is the partition function, and the energy function $E(\boldsymbol{v})$ is typically defined by a chosen Markov random field (MRF). For example, a BM [21] is a binary MRF, where binary units update their state based on a bias and pairwise connections to other units. A MRF where all variables $\boldsymbol{v}$ are driven by data is fully visible; otherwise, the N' units $\boldsymbol{h} = \{h_1, \ldots, h_{N'}\}$ which are not driven by data are denoted as hidden.

A restricted Boltzmann machine (RBM) [32] is a BM in which hidden and visible units are organized into layers, where a layer is defined by the property that there are no interactions among units in the same layer. For example, a typical energy function for an RBM is of the form:

$$E(\boldsymbol{v}, \boldsymbol{h}, \boldsymbol{\theta}) = -\sum_{i=1}^{N} b_i v_i - \sum_{j=1}^{N'} b'_j h_j - \sum_{\{i,j\}} W_{i,j} v_i h_j, \tag{22}$$

where the summation $\{i, j\}$ is determined by the graph edges and $\boldsymbol{\theta}$ is the vector of length K of all interaction parameters in the graph. This defines a joint distribution over $\boldsymbol{v}$ and $\boldsymbol{h}$:

$$\tilde{p}(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})}\exp[-E(\boldsymbol{v}, \boldsymbol{h}, \boldsymbol{\theta})]. \tag{23}$$

Each parameter θ_k in this MaxEnt distribution controls a corresponding moment $\tilde{\mu}_k$, given by $\tilde{\mu}_k = \partial \ln Z(\boldsymbol{\theta})/\partial\theta_k$.

Define a *dynamic Boltzmann distribution* as one with time-dependent interaction parameters:

$$\tilde{p}(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}(t)) = \frac{1}{Z[\boldsymbol{\theta}(t)]}\exp\{-E[\boldsymbol{v}, \boldsymbol{h}, \boldsymbol{\theta}(t)]\}. \tag{24}$$

For example, the energy function of the RBM becomes

$$E[\boldsymbol{v}, \boldsymbol{h}, \boldsymbol{\theta}(t)] = -\sum_{i=1}^{N} b_i(t) v_i - \sum_{j=1}^{N'} b'_j(t) h_j - \sum_{\{i,j\}} W_{i,j}(t) v_i h_j. \tag{25}$$

This is a specific case of a spatial dynamic Boltzmann distribution (1) in the discrete lattice limit. To see this, assign to every visible unit v_i a spatial location x_i. By taking self-interaction functions $\nu_1(x, t) = -\sum_i b_i(t)\delta_{x,x_i}$ in (1), we recover the first term in (25) with $v_i \in \{0, 1\}$, where δ_{x,x_i} is unity if the coordinates are coincident and zero otherwise.

Similarly, hidden units can also be represented in continuous space. Let the species labels α_v denote visible units and β_h denote hidden units, and assign to every hidden unit h_j a spatial location y_j. The weights between layers are then obtained by taking pairwise interactions $\nu_2(\alpha, \beta, x, y, t) = -\sum_{\{i,j\}} W_{i,j}(t)\delta_{x,x_i}\delta_{y,y_j}\delta_{\alpha,\alpha_v}\delta_{\beta,\beta_h}$.

A. An adjoint method learning problem for restricted Boltzmann machines

Introduce for each interaction parameter θ_k, $k = 1, \ldots, K$, in the interaction graph a *time-evolution function* F_k forming an autonomous ODE system [analogously to (17)]:

$$\frac{d}{dt}\theta_k(t) = F_k(\boldsymbol{\theta}(t)), \tag{26}$$

with initial conditions $\theta_k(t_0) = \theta_{k,0}$. To obtain from the variational problem derived in Sec. II B an ordinary optimization problem for parameters, further consider the paramaterization by the vectors $\boldsymbol{u}_k$ of size M_k, generally unique for every k:

$$\frac{d}{dt}\theta_k(t) = F_k(\boldsymbol{\theta}(t); \boldsymbol{u}_k). \tag{27}$$

Analogously to the continuous case, define as the objective function the KL divergence between the true and reduced models, p and $\tilde{p}$, over all times [analogously to (4)]:

$$S = \int_{t_0}^{t_f} dt\ \mathcal{D}_{\mathrm{KL}}(p||\tilde{p}),$$
$$\mathcal{D}_{\mathrm{KL}}(p||\tilde{p}) = \sum_{\boldsymbol{z}} p(\boldsymbol{z}) \ln \frac{p(\boldsymbol{z})}{\tilde{p}(\boldsymbol{z}; \{\boldsymbol{u}\})}. \tag{28}$$

where $\{\boldsymbol{u}\} = \{\boldsymbol{u}_k\}_{k=1}^K$. Minimizing S is thus equivalent to maximizing the log-likelihood of the observed data given the parameters, i.e., $L(\{\boldsymbol{u}\}; \boldsymbol{z}) = \log\tilde{p}(\boldsymbol{z}; \{\boldsymbol{u}\})$. A more common approach is to instead maximize the conditional likelihood of observations conditioned on the first observation: $L(\{\boldsymbol{u}\}; z_2, z_3, \ldots|z_1) = \log\tilde{p}(z_2, z_3, \ldots|z_1; \{\boldsymbol{u}\})$ or

Algorithm 1 Stochastic Gradient Descent for Learning Restricted Boltzmann Machine Dynamics.

Initialize
  Parameters $\boldsymbol{u}_k$ controlling the functions $F_k(\boldsymbol{\theta};\boldsymbol{u}_k)$ for all $k = 1,\ldots,K$.
  Time interval $[t_0, t_f]$, a formula for the learning rate λ.
while not converged **do**
  Initialize $\Delta F_{k,i} = 0$ for all $k = 1,\ldots,K$ and parameters $i = 1,\ldots,M_k$.
  for sample in batch **do**
    ▷ *Generate trajectory in reduced space* $\boldsymbol{\theta}$*:*
    Solve the PDE constraint (27) for $\theta_k(t)$ with a given IC $\theta_{k,0}$ over $t_0 \leqslant t \leqslant t_f$, for all k.
    ▷ *Wake phase:*
    Evaluate moments $\mu_k(t)$ of the data for all k, t.
    ▷ *Sleep phase:*
    Evaluate moments $\tilde{\mu}_k(t)$ of the Boltzmann distribution.
    ▷ *Solve the adjoint system:*
    Solve the adjoint system (31) for $\phi_k(t)$ for all k, t.
    ▷ *Evaluate the objective function:*
    Update $\Delta F_{k,i}$ as the cumulative moving average of the sensitivity equation (30) over the batch.
  ▷ *Update to decrease objective function:*
  $u_{k,i} \to u_{k,i} - \lambda \Delta F_{k,i}$ for all k, i.

similar causal relations. For Markov chains, this approach is highly successful (leading to, e.g., Kalman filters; see Ref. [33] for an introduction). If a prior is available, then Bayesian methods that compute the posterior $\tilde{p}(\{\boldsymbol{u}\};\boldsymbol{z}) \propto \tilde{p}(\boldsymbol{z};\{\boldsymbol{u}\}) \times \tilde{p}(\{\boldsymbol{u}\})$ can provide further improvements. The advantage of the current approach is that a reduced physical model can be enforced through the parametrization (27). This model can be based on prior information, such as reaction networks with known solutions [22]. A second advantage is that the generalization to spatially continuous systems follows naturally using PDEs as in (8).

The time integral in S can lead to undesired extrema, for example for periodic systems where the objective function may not minimize the KL divergence at each time point. One algorithmic strategy for eliminating these in practice is to shift the limits of integration during the optimization, as done in the examples of Sec. IV A.

Minimizing the objective function defines a PDE-constrained optimization problem: minimize (28) subject to the PDE constraint (27). Define the Lagrangian function [analogously to (9)]:

$$\mathcal{L}(t;\{\boldsymbol{u}\}) = \mathcal{D}_{\mathrm{KL}}(p||\tilde{p}) + \sum_{k=1}^{K} \phi_k(t) \times \left\{ \frac{d}{dt}\theta_k(t) - F_k[\boldsymbol{\theta}(t);\boldsymbol{u}_k] \right\}, \tag{29}$$

where we have introduced the adjoint variables ϕ_k associated with each θ_k. Taking the derivative of the objective function $S = \int_{t_0}^{t_f} dt\ \mathcal{L}(t;\{\boldsymbol{u}\})$ with respect to a parameter gives the sensitivity equation [analogously to (18)]:

$$\frac{dS}{du_{k,i}} = -\int_{t_0}^{t_f} dt\ \frac{\partial F_k[\boldsymbol{\theta}(t);\boldsymbol{u}_k]}{\partial u_{k,i}} \phi_k(t), \tag{30}$$

and taking the derivative with respect to θ gives the ODE system obeyed by the adjoint variables [analogously to (19)]:

$$\frac{d}{dt}\phi_k(t) = \tilde{\mu}_k(t) - \mu_k(t) - \sum_{l=1}^{K} \frac{\partial F_l[\boldsymbol{\theta}(t);\boldsymbol{u}_l]}{\partial \theta_k(t)} \phi_l(t), \tag{31}$$

where $\mu_k(t')$ and $\tilde{\mu}_k(t')$ are averages taken over to p and $\tilde{p}$ at time t', and the boundary condition is $\phi_k(t_f) = 0$.

Algorithm I outlines how this optimization problem can be solved in practice. The inner loop of an "wake" and "sleep" phase of sampling are identical to that of BM learning. Standard algorithmic improvements are possible, such as the use of accelerated gradient descent methods such as Adam [34], and using persistent contrastive divergence (PCD) [35] to estimate the moments of the reduced model $\tilde{\mu}_k(t')$.

Adjoint methods for solving PDE-constrained optimization problems are also called "black-box" methods [36,37], since the PDE constraint (27) is eliminated in the derivation of the sensitivity equation (30). A competing class of methods (sometimes referred to as "all-at-once" methods) treat the constraint explicitly in the optimization, and may offer a computational advantage over this approach. These include sequential quadratic programming and augmented Lagrangian methods.

Additional constraints or regularization terms can be included in the optimization, such as conserved quantities identified from the left null space of the net stoichiometry matrix. For example, L_2 regularization can be incorporated into the objective function:

$$S = \int_{t_0}^{t_f} dt\ \mathcal{D}_{\mathrm{KL}}(p||\tilde{p}) + \lambda_r \int_{t_0}^{t_f} dt\ \sum_{k=1}^{K} [\theta_k(t) - \overline{\theta}_k(t)]^2, \tag{32}$$

where $\overline{\theta}_k(t)$ are some specified functions or otherwise constant and λ_r is a regularization parameter. In this case, the adjoint variables are given by:

$$\frac{d}{dt}\phi_k(t) = \tilde{\mu}_k(t) - \mu_k(t) + 2\lambda_r[\theta_k(t) - \overline{\theta}_k(t)] - \sum_{l=1}^{K} \frac{\partial F_l(\boldsymbol{\theta}(t);\boldsymbol{u}_l)}{\partial \theta_k(t)} \phi_l(t). \tag{33}$$

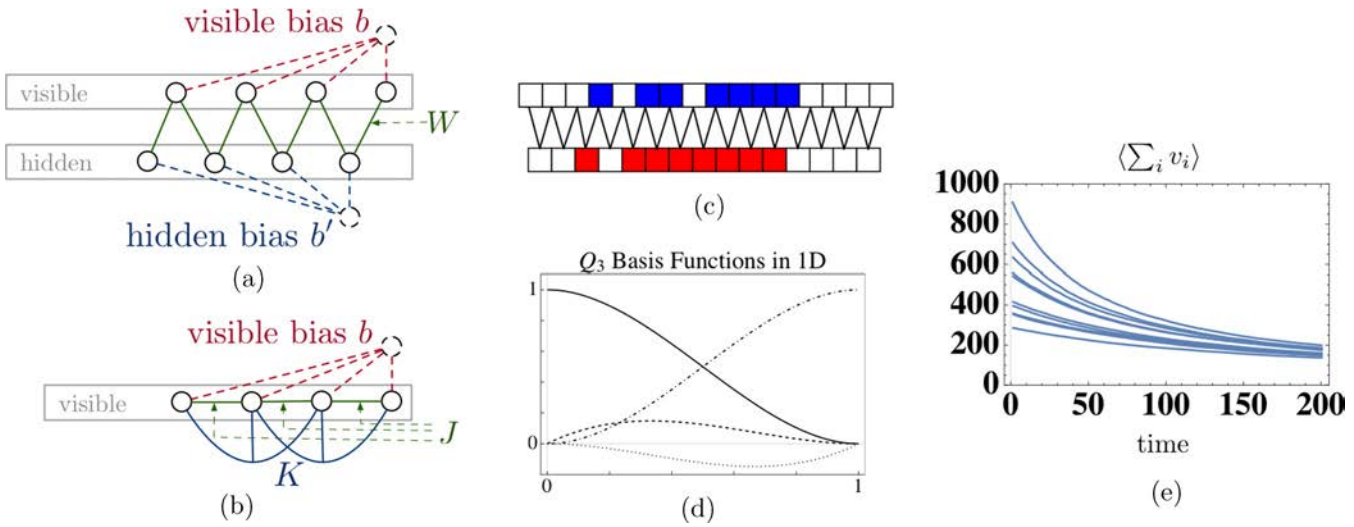

FIG. 1. Comparison of a fully visible and a latent variable model for capturing local correlations in a 1D lattice. (a) One-dimensional lattice with one hidden layer (similar to an RBM). Note that in this simplified example, W is a single translation invariant parameter rather than a matrix as common in RBMs. (b) Fully visible model for a 1D lattice including NN interactions J and NNN interactions K. (c) An example state of the hidden layer model, where blue indicates the presence of a particle in the visible layer and likewise red for the hidden layer. By learning the parameters, the hidden layer can be tuned to capture the presence of NNs. (d) The basis functions of the Q_3 family of C_1 finite elements in 1D (Hermite polynomials), used to parametrize the right-hand sides of (38) and (40). Basis functions in higher dimensions are constructed as tensor products of the 1D polynomials. (e) Moments of stochastic simulations for 10 of the 50 initial conditions used for training (each trajectory obtained from averaging over 50 lattices simulated from the same initial condition).

B. Finite-element parameterization

What choice should be made for the parametrization (27) of the right-hand sides of the differential equations? In Ref. [22], we considered simple reaction-diffusion systems from which general forms of approximate models could be inferred that maintain physical interpretations. A second approach also explored in Ref. [22] is to use a separate moment closure approximation to derive analytic solutions for simple reaction systems on 1D lattices, where the inverse Ising problem is analytically solvable. The form of (27) can then be taken as either linear or nonlinear combinations of known solutions.

Here, we take a *finite-element method* [38] approach to the parametrization that is more aligned with the unsupervised learning problem in a Boltzmann machine. The space of solutions to the general variational problem (16), which is some Banach space, is therefore restricted to the space of finite-element method solutions.

An important restriction is that the learning rule (30) requires C_1 finite elements. One choice for such elements is the Q_3 family of finite elements [39], which has the advantage that basis functions in dimensions higher than one are easily constructed as tensor products of 1D cubic polynomials.[2] For C_1 elements that control the value of the function and its derivative at the endpoints, these polynomials are just the Hermite polynomials, shown in Fig. 1(d).

We introduce for each time-evolution function in (27) a domain of hypercubic cells, with 4^d degrees of freedom, where d are the number of arguments to F_k. In practice, we found it is rarely necessary to have more than $d = 3$ arguments (see Sec. IV). For $d = 3$, each cube has 64 degrees of freedom (8 degrees of freedom at each vertex, specifying the function value and derivatives). For a cubic lattice of $V = L_1 \times L_2 \times L_3$ cells, there are $8V$ degrees of freedom in total, with the parametrization taking the usual form in terms of the basis functions f_l associated with each degree of freedom:

$$F_k(\theta_1, \theta_2, \theta_3; \boldsymbol{u}_k) = \sum_{l=1}^{8V} u_l f_l(\theta_1, \theta_2, \theta_3). \tag{34}$$

Note that here the right-hand side of the differential equation is parameterized (as opposed to the solution of the differential equation), since the objective of the learning algorithm is to determine a suitable differential equation model.

IV. LEARNING REACTION-DIFFUSION SYSTEMS ON LATTICES

Recall that the state of a reaction-diffusion system at some time is described by n particles of species $\boldsymbol{\alpha}$ located at positions $\boldsymbol{x}$ in generally continuous 3D space. To make an explicit connection to binary random variables, we consider a simpler model of particles hopping on a discrete lattice in the single-occupancy limit. To generate stochastic simulations of such a system, we adapt the method of Takayasu and Tretyakov [40] for a lattice-based variant of the popular Gillespie SSA [16] as follows: At each time step:

[2] An alternative choice for tetrahedral meshes is the P_3 family of finite elements.

(1) Perform unimolecular reactions following the standard Gillespie SSA.

(2) Iterate over all particles in random order; for each:

(a) Hop to a neighboring site, chosen at random with equal probability.

(b) If the site is unoccupied, then the move is accepted. If the site is occupied, then a bimolecular reaction occurs with some probability; else, the move is rejected and the particle is returned to the original site.

The lattice on which particles hop is designated as the visible part of the MRF. Assign a unique index i to each of the N sites in the lattice, and let the vector of possible species be $\boldsymbol{s}$ of size M in some arbitrary ordering (excluding $\varnothing$ to denote an empty site). Spins at a site i are now multinomial units, represented as a vector $\boldsymbol{v}_i$ of length M where entries $v_{i,\alpha} \in \{0, 1\}$ for $\alpha = 1, \ldots, M$ denote the absence or presence of a particle of species s_α (an n-vector model in statistical mechanics). The single-occupancy limit corresponds to the implicit constraint that the vectors are of unit length, i.e., $\sum_{\alpha=0}^{M} v_{i,\alpha} = 1$, where $\alpha = 0$ denotes an empty site. The matrix $\boldsymbol{V}$ of size $N \times M$ describes the state of the visible part of the MRF, where each row denotes a lattice site.

Likewise, introduce hidden layer species $\boldsymbol{s}'$ of size M', which may be different from $\boldsymbol{s}$. Indexing all hidden sites as $j = 1, \ldots, N'$, hidden unit vectors are $\boldsymbol{h}_j$ of length M'. The state of the hidden units is $\boldsymbol{H}$ of size $N' \times M'$, with the single-occupancy constraint as before.

The dynamic Boltzmann distribution becomes $p[\boldsymbol{V}, \boldsymbol{H}|\boldsymbol{\theta}(t)] = \exp\{-E[\boldsymbol{V}, \boldsymbol{H}, \boldsymbol{\theta}(t)]\}/Z[\boldsymbol{\theta}(t)]$, where interaction parameters $\boldsymbol{\theta}(t)$ may also be species dependent. For example, the energy function for the RBM becomes

$$\begin{aligned} E[\boldsymbol{V}, \boldsymbol{H}, \boldsymbol{\theta}(t)] = &- \sum_{i=1}^{N}\sum_{\alpha=1}^{M} b_{i,\alpha}(t) v_{i,\alpha} \\ &- \sum_{j=1}^{N'}\sum_{\beta=1}^{M'} b'_{j,\beta}(t) h_{j,\beta} \\ &- \sum_{\{i,j\}}\sum_{\alpha,\beta} W_{i,j,\alpha,\beta}(t) v_{i,\alpha} h_{j,\beta}. \end{aligned} \tag{35}$$

A. Learning hidden layers for moment closure

A typical problem in many-body systems is the appearance of a hierarchy of moments, where the time evolution of a given moment depends on higher-order moments. Moment closure approximations terminate this infinite hierarchy at some finite order. In this section, we develop the perspective of the learning problem (30) as a closure approximation using a simple pedagogical example. We note some similarity to previously proposed closure schemes [14,15], as well as to entropic matching [13], although the current approach differs in the objective function (28) and the formulation for spatially continuous systems in Sec. II.

Consider a bimolecular-annihilation process on a 1D lattice of length N, where particles of a single species A hop and react according to $A + A \to \varnothing$. The time evolutions of the first two moments are (see Appendix B)

$$\begin{aligned} \frac{d}{dt}\left\langle \sum_i v_i \right\rangle &= -2k_r \left\langle \sum_i v_i v_{i+1} \right\rangle, \\ \frac{d}{dt}\left\langle \sum_i v_i v_{i+1} \right\rangle &= 2D\left\langle \sum_i v_i v_{i+2} \right\rangle - 2k_r \left\langle \sum_i v_i v_{i+1} v_{i+2} \right\rangle \\ &\quad + (k_r - 2D)\left\langle \sum_i v_i v_{i+1} \right\rangle, \end{aligned} \tag{36}$$

where k_r is the reaction rate and D the diffusion rate. The simplest graph to capture such observables is a fully visible Markov random field with N units, i.e., a 1D Ising model including interactions up to some order. For example, including third-order interactions, let:

$$\begin{aligned} E[\boldsymbol{v}, b(t), J(t), K(t)] = &-b(t)\sum_{i=1}^{N} v_i - J(t)\sum_{i=1}^{N-1} v_i v_{i+1} \\ &- K(t)\sum_{i=1}^{N-2} v_i v_{i+1} v_{i+2}, \end{aligned} \tag{37}$$

where b is the bias, J is the nearest neighbor (NN) interaction term, and K is the next-nearest-neighbor (NNN) interaction term. Let the differential equation model be

$$\begin{aligned} \dot{b} &= F_b(b, J, K; \boldsymbol{u}_b), \\ \dot{J} &= F_J(b, J, K; \boldsymbol{u}_J), \\ \dot{K} &= F_K(b, J, K; \boldsymbol{u}_K), \end{aligned} \tag{38}$$

for some parameter vectors $\boldsymbol{u}$ to be learned, where time derivatives are denoted as $\dot{x} = d/dt$. The corresponding graphical model is illustrated in Fig. 1(b). The choice of the energy function in (37) defines which moments are explicitly captured by the reduced model. The additional choice of the form of the differential equations F_γ defines the moment closure approximation made.

We next show through computational experiments that the introduction of hidden layers can improve on a fully visible closure model:

(1) In any closure scheme, moments beyond a certain order are not captured explicitly by the model, so that their approximation may be poor. The representation power of hidden layers [24] can be used to incorporate information about which higher-order moments are relevant to the data set.

(2) Two distinct states having the same lower-order moments are indistinguishable in the reduced model (the model is not sufficiently high dimensional). Hidden layers may be able to separate such states if their connectivity is suitably chosen to represent relevant higher-order correlations, even if the model remains low order.

(3) The number of higher-order terms appearing on the right of (36) grows with the order on the left. This problem is compounded if species labels are included. Hidden layers and a restriction on the number of species M' allowed to occupy hidden units may be used to approximate such higher-order interactions with fewer parameters.

It is generally difficult to choose the optimal close approximation, i.e., to know which moments are relevant to the

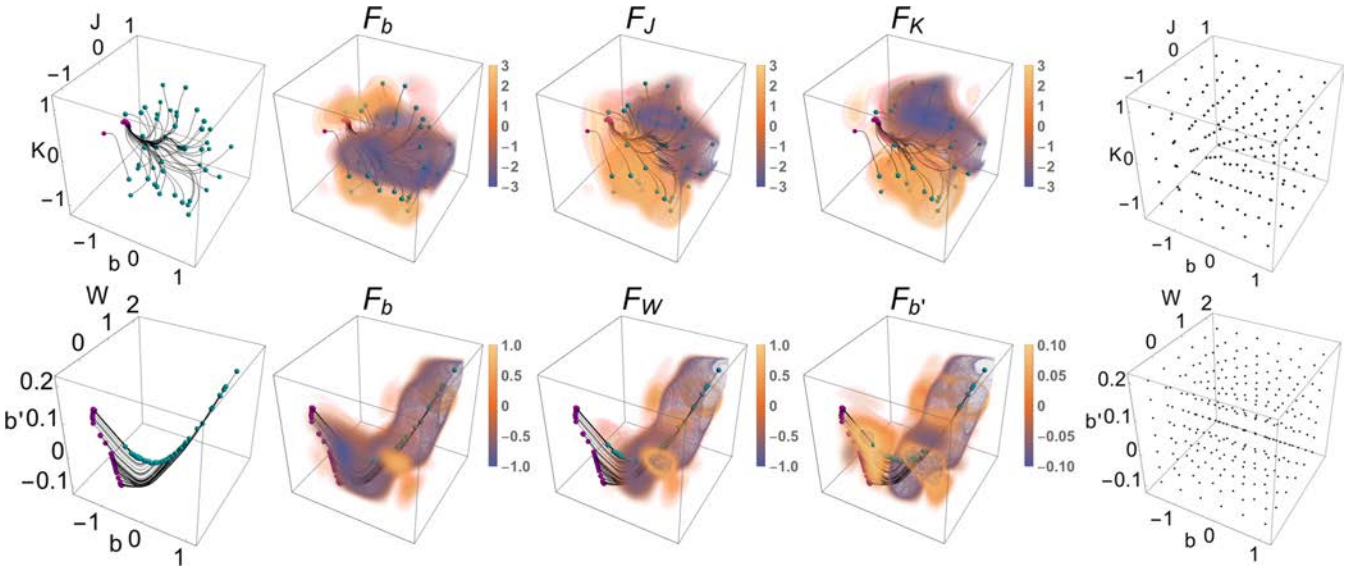

FIG. 2. Top row: Learned time-evolution functions for the fully visible model (38), using the Q_3, C_1 finite-element parametrization (34) with cells of size $0.5 \times 0.5 \times 0.5$ in (b, J, K). Left panel: Training set of initial points (b, J, K) (cyan) sampled evenly in $[-1, 1]$. Stochastic simulations for each initial point are used as training data (learned trajectories shown in black, endpoints in magenta). Middle three panels: The time evolution functions learned, where the heat map indicates the value of F_γ in (38). Right panel: Vertices of the finite-element cells used. Bottom row: Hidden layer model (40) and parametrization (34) with cells of size $0.5 \times 0.5 \times 0.05$ in (b, W, b'). Initial points are generated by BM learning applied to the points of the visible model. Note that the coefficients corresponding to the other seven degrees of freedom at each vertex are also learned (not shown), i.e., the first derivatives in each parameter.

time evolution of a given data set. A key advantage of the present approach is that the connectivity of the hidden layers may be chosen based on the differential equations derived from the chemical master equation. For example, consider to the bimolecular annihilation system (36): If the goal is to accurately model the mean number of particles, then the right-hand side of (36) shows that the nearest-neighbor moment is relevant to the time evolution. The graphical model of the reduced system could therefore introduce a hidden unit for every pair of neighboring lattice sites ($N-1$ units in the hidden layer), with corresponding energy function:

$$E[\boldsymbol{v}, \boldsymbol{h}, b(t), W(t), b'(t)] = -b(t)\sum_{i=1}^{N} v_i - b'(t)\sum_{j=1}^{N-1} h_j - W(t)\sum_{j=1}^{N-1}\sum_{i\in\{j,j+1\}} v_i h_j, \quad (39)$$

where b is bias for visible units, b' is the bias for hidden units, and W are the weights connecting visible and hidden units. Let the differential equation model be

$$\begin{aligned} \dot{b} &= F_b(b, b', W; \boldsymbol{u}_b), \\ \dot{b}' &= F_{b'}(b, b', W; \boldsymbol{u}_{b'}), \\ \dot{W} &= F_W(b, b', W; \boldsymbol{u}_W). \end{aligned} \quad (40)$$

The corresponding graphical model is shown in Figs. 1(a) and 1(c).

The time-evolution functions for (38) and (40) are learned using Algorithm 1 and compared in Fig. 2. For the visible model, cells of size $0.5 \times 0.5 \times 0.5$ in (b, J, K) are used, and for the hidden layer model cells of size $0.5 \times 0.5 \times 0.05$ in (b, W, b'), as shown in Fig. 2.

As training data, 50 points (b, J, K) are sampled evenly over $(b, J, K) \in [-1, 1]^3$. Each point corresponds to an initial distribution (37), from each of which 50 lattices of length $N = 1000$ are sampled (top left panel of Fig. 2). The corresponding initial conditions in (b, W, b') space are learned separately using the BM learning algorithm (bottom left panel of Fig. 2). Each lattice is simulated for 200 time steps of size $\Delta t = 0.01$ with reaction probability $p_r = 0.01$ on encounters for the reaction $A + A \to \varnothing$, as shown in Fig. 1(e). These trajectories are pooled for Algorithm 1. Note that a single set of parameter vectors $\{\boldsymbol{u}\}$ in (38) and (40) is learned, i.e., the parameter vectors are shared among trajectories from all initial conditions.

For the fully visible model, sleep phase moments are estimated by running a Gibbs sampler for a single step. Similarly, for the hidden model, wake and sleep phase moments are estimated by a single step of contrastive divergence (CD), i.e., CD-1. The learning rate used in both models is $\lambda = 1$ for 200 optimization steps.

The time integral in the action (28) can lead to undesired extrema, e.g., for periodic trajectories. We use an online algorithm to shift the limits of integration in (30) as new data are available:

$$\frac{dS}{du_{k,i}} = \int_{\tau}^{\tau+\Delta\tau} dt\ \frac{\partial F_k(\boldsymbol{\theta}(t); \boldsymbol{u}_k)}{\partial u_{k,i}} \phi_k(t), \quad (41)$$

where $\Delta\tau$ is fixed and τ is gradually incremented $t_0 \leqslant \tau \leqslant t_f - \Delta\tau$. In this case, the PDE constraint (27) is solved from t_0 to τ, decreasing the size of the trajectories early in the training. Further, the adjoint system (31) only has to be solved backward from $\phi(\tau + \Delta\tau) = 0$ to $\phi(\tau)$, which also

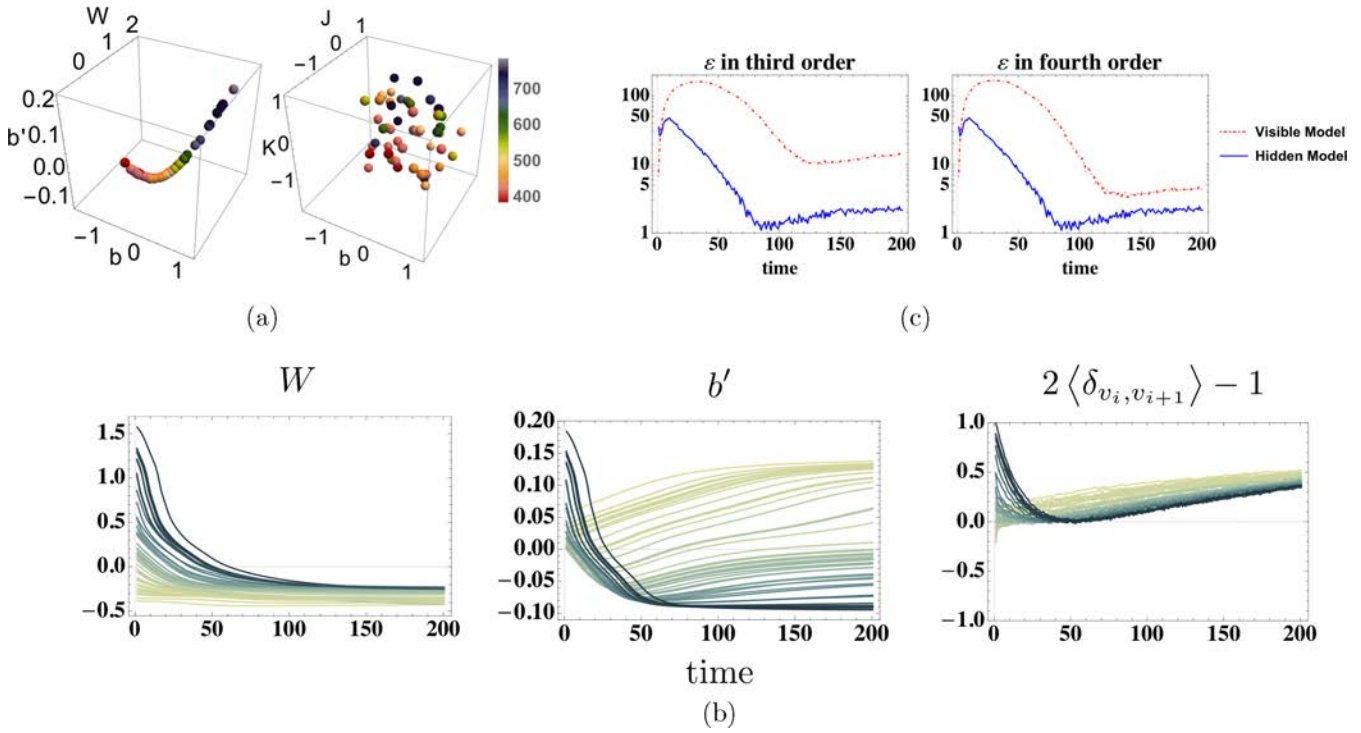

FIG. 3. (a) NN moment $\langle \sum_i v_i v_{i+1} \rangle$ of the two models. The more compact representation learned by the hidden layer model (left) captures low range spatial correlations, while the fully visible model (right) shows no apparent organization. (b) The parameters W and b' for the hidden layer model for the 50 initial conditions (b is monotonically decreasing for all trajectories). The learned parameters encode the spatial correlation $2\langle \delta_{v_i v_{i+1}} \rangle - 1$ shown on the right. This shows the moment closure approximation learned by the reduced model (see text). (c) RMSE in the third-order moment $\langle \sum_i v_i v_{i+1} v_{i+2} \rangle$ and fourth-order moment $\langle \sum_i v_i v_{i+1} v_{i+2} v_{i+3} \rangle$, calculated from a set of test trajectories (not shown). Both models reproduce the observables with reasonable accuracy, however, the error in the hidden layer model is lower due to the more compact representation learned.

controls the magnitude of the update steps as the length of the trajectory grows, allowing a constant learning rate to be used. For the annihilation system, we found that fixing $\Delta\tau = 5$ time steps and shifting $\tau \to \tau + 1$ every two optimization steps gave fast convergence.

Figure 2 shows the learned time-evolution functions and trajectories of the training data. For the visible model, these show an expected symmetric structure. As particles diffuse and NN and NNN moments decay, F_J and F_K force $J, K \to 0$ everywhere, while the bias term tends to negative infinity. The representation learned by the hidden layer model is more compact. Figure 3(a) shows the nearest-neighbor moment $\langle \sum_i v_i v_{i+1} \rangle$ overlaid onto the initial conditions, showing an almost monotonic organization from low to high values by which the model can distinguish these states (no organization is apparent in the visible model). Figure 3(b) shows the learned parameter trajectories: b monotonically decreases (not shown), W asymptotically approaches a negative value, and b' either increases monotonically or initially decreases before increasing again. This division corresponds to the decay of spatial correlations $2\langle \delta_{v_i v_{i+1}} \rangle - 1$ (such that 1 corresponds to a fully correlated lattice and -1 to a fully anticorrelated lattice), also shown in Fig. 3(b). The two types of trajectories of b' have a clear correspondence to two types of trajectories in the correlation function, and the separation is visible in $F_{b'}$ in the negative and positive regimes. We conclude that the moment closure approximation learned by the model therefore captures relevant low-range spatial correlations to approximate the right-hand sides of the moment equations (36) identified from the CME.

To assess the accuracy of the reduced models, we generate a test set of points (b, J, K) and the learn the corresponding points (b, W, b') as before. These are evolved in time using the learned DE systems (38) and (40). Define $\varepsilon(t) = \sqrt{\langle [\mu(t) - \tilde{\mu}(t)]^2 \rangle}$ as the root-mean-square error (RMSE) between some moments of the reduced model $\tilde{\mu}$ and the stochastic simulations μ, where the moments are approximated by averaging over 50 samples. Figure 3(c) shows the RMSE for the third-order moment $\langle \sum_i v_i v_{i+1} v_{i+2} \rangle$ and fourth-order moment $\langle \sum_i v_i v_{i+1} v_{i+2} v_{i+3} \rangle$. Both models have relatively low error in reproducing the observables, however, the error in the hidden layer model is lower than in the visible model. This is because the representation learned by the hidden layer model is more compact, in that states initially distributed uniformly in (b, J, K) space are mapped to an approximately 1D curve in (b, W, b') space. Yet higher accuracies may be possible by further tailoring that parametrizations of the differential equations from the cubic finite elements used here.

B. Learning the Rössler oscillator

The Williamowski-Rössler oscillator system [41] is a chemical version of a spiral oscillator in three species. The original formulation requires additional species that are fixed

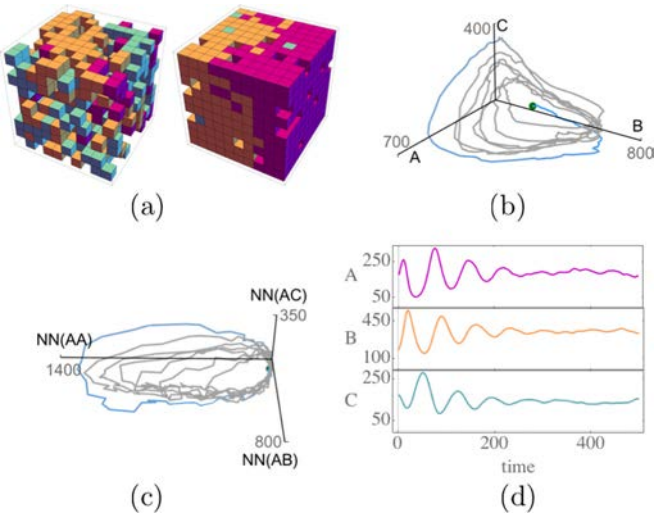

FIG. 4. Rössler oscillator on a 3D lattice. (a) Snapshots of a stochastic simulation on a $10 \times 10 \times 10$ lattice (A, B, and C in pink, orange, and cyan). (b) Moments from a single simulation over 500 time steps, producing a stochastic version of the characteristic attractor of the well-known deterministic model. (c) Nearest-neighbor moments in the simulation of (b) show similar structure. (d) Relaxation to a stationary distribution, indicated by the convergence of the means from averaging over 300 stochastic simulations.

at constant concentration. Recent work [42], however, has developed a volume-excluding version where these constraints are incorporated into pseudo-first-order reaction rates, eliminating the need for additional reservoir populations. We follow this approach, such that the reaction system for species A, B, and C is

$$A \underset{p_1}{\overset{k_1}{\rightleftharpoons}} 2A, \quad A + B \xrightarrow{p_2} 2B, \quad A + C \xrightarrow{p_3} \varnothing,$$
$$B \xrightarrow{k_2} \varnothing, \quad C \underset{p_4}{\overset{k_3}{\rightleftharpoons}} 2C, \tag{42}$$

where the unimolecular reaction rates used are $k_1 = 30, k_2 = 10, k_3 = 16.5$ (arbitrary units), and the probabilities for bimolecular reactions are $p_1 = 0.1$, $p_2 = 0.4$, $p_3 = 0.24$, $p_4 = 0.36$. We simulate this system on a 3D lattice of size $10 \times 10 \times 10$ sites in the single-occupancy limit as before. Figure 4 shows snapshots of such a stochastic simulation. Figure 4(b) in particular shows the characteristic shape of the Rössler oscillator, with further structures evident in higher-order moments shown in Fig. 4(c). A snapshot of the spatial waves that occur during transitions between A-, B-, and C-dominated regimes is shown in Fig. 4(a).

The time evolution of the mean number of particles in A, B, and C, denoted by μ_α, is related to the number of nearest neighbors, denoted by $\Delta_{\alpha\beta}$, as follows (see Appendix B for derivation):

$$\frac{d}{dt}\mu_A = k_1\mu_A - \kappa_1\Delta_{AA} - \kappa_2\Delta_{AB} - \kappa_3\Delta_{AC},$$
$$\frac{d}{dt}\mu_B = \kappa_2\Delta_{AB} - k_2\mu_B, \tag{43}$$
$$\frac{d}{dt}\mu_C = -\kappa_3\Delta_{AC} + k_3\mu_C - \kappa_4\Delta_{CC},$$

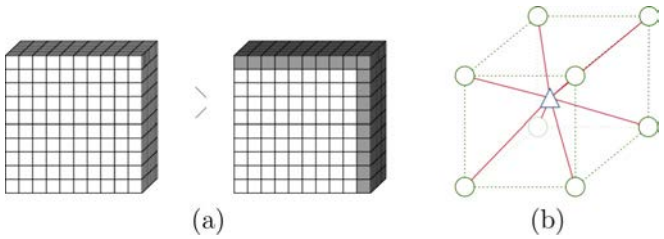

FIG. 5. (a) Graph to learn for the Rössler oscillator. The lattice on the left corresponds to the visible layer, equivalent to the $10 \times 10 \times 10$ cube in Fig. 4; the right corresponds to the hidden layer. Gray units in the hidden layer denote those units which implement periodic boundary conditions to the visible layer. (b) Connectivity of hidden layer. Each cube of eight neighboring units in the visible layer (green circles) is connected to a single unit (blue triangle) in the hidden layer (connections shown in red), resembling a body-centered cubic structure. Biases for the units are not shown.

where κ_1, κ_2, κ_3, and κ_4 are the reaction rates for the bimolecular reactions specified by probabilities p_1, p_2, p_3, and p_4 above. As previously, this system is not closed, such that two close initial states in Fig. 4(b) will diverge over their long-term time evolution. The challenge for the latent variables in the reduced differential equation model is to incorporate relevant higher-order correlations to separate states which are close in their lower-order moments.

As in Sec. IV A, let the visible part of the graph be the lattice of Fig. 4(a). For the hidden layer, we choose a connectivity that coarse grains the visible lattice by one unit in each spatial dimension as shown in Fig. 5. Note that the hidden layer is also of size $10 \times 10 \times 10$ units that implement periodic boundary conditions. The visible layer of the graph is multinomial in one of $\{A, B, C, \varnothing\}$, and similarly the hidden layer in $\{X, Y, Z, \varnothing\}$. The corresponding energy model is

$$E(\boldsymbol{V}, \boldsymbol{H}, \boldsymbol{\theta}(t)) = -\sum_i \sum_{\alpha\in\{A,B,C\}} b_\alpha v_{i,\alpha} - \sum_j \sum_{\alpha\in\{X,Y,Z\}} b_\alpha h_{j,\alpha} - \sum_{\{i,j\}} \left(W_{AX} v_{i,A} h_{j,X} + W_{BY} v_{i,B} h_{j,Y} + W_{CZ} v_{i,C} h_{j,Z}\right), \tag{44}$$

where $\boldsymbol{H}$ refers to the hidden layer and the sum over $\{i, j\}$ implements the connectivity shown in Fig. 5 and

$$\dot{\gamma} = F_\gamma(b_A, b_B, b_C; \boldsymbol{u}_\gamma) \tag{45}$$

for $\gamma \in \{b_A, b_B, b_C, W_{AX}, W_{BY}, W_{CZ}, b_X, b_Y, b_Z\}$. The right-hand side of the differential equation is parameterized (34) by cubic C_1 finite elements as before. To reduce the complexity of the model, we have purposefully omitted interactions $W_{AY}, W_{AZ}, W_{BX}, W_{BZ}, W_{CX}, W_{CY}$. With this choice, the latent species X coarse grains the visible species A, and similarly for Y, B and C, Z. Note that all differential equation models share the same domain in (b_A, b_B, b_C) space. While the biases h_A, h_B, h_C are the Lagrange multipliers corresponding to the constraints for the number of particles of each species, through the energy function (44) both biases *and* weights together control all spatial correlations of the model.

Stochastic simulations are generated from an initial state with $b_A = b_B = b_C = -\ln(2)$, $W_{AX} = W_{BY} = W_{CZ} = W_{XY} =$

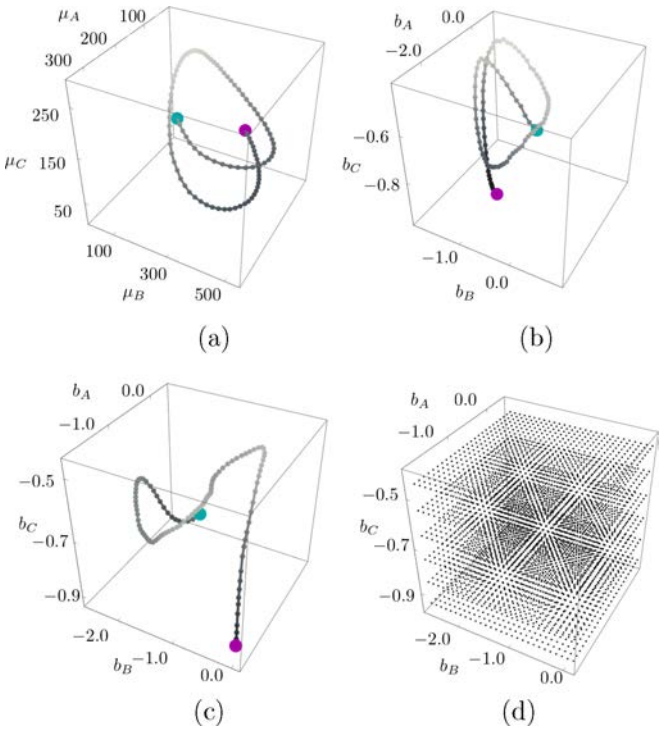

FIG. 6. (a) The first 100 time steps of the mean number of A, B, and C in the Rössler oscillator system. (b) Interaction parameters for a MaxEnt model constrained on the moments in (a) given by Eq. (46). (c) The learned trajectory of (44) in (b_A, b_B, b_C) space, with initial condition $[-\ln(2), -\ln(2), -\ln(2)]$. The bias parameters have been tuned to control both the means and spatial correlations, together with the weights (not shown). Grayscale value indicates b_C component for clarity, scaled from dark $[\min(b_C)]$ to light $[\max(b_C)]$. Initial point is shown in cyan, and endpoint in magenta. (d) Vertices of the finite-element cells of side length 0.1 used to parametrize the differential equations (45).

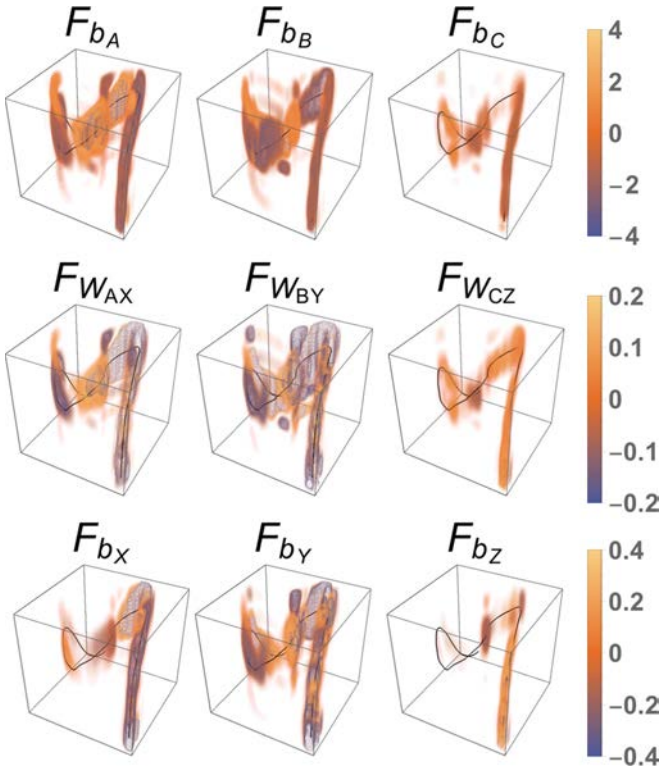

FIG. 7. Learned time-evolution functions (45) in (b_A, b_B, b_C) space [see Fig. 6(d) for the vertices used], and the resulting trajectory in black [see Fig. 6(c)].

$W_{YZ} = 0$, and $b_X = b_Y = b_Z = -\ln(1/7)$. By setting the initial weights to zero, this is the MaxEnt state given that the number of particles is $\mu_A = \mu_B = \mu_C = 200$, since with zero weight:

$$\mu_\alpha = 1000 \frac{e^{b_\alpha}}{1 + \sum_{\beta=A,B,C} e^{b_\beta}} \tag{46}$$

for $\alpha \in \{A, B, C\}$, and where the factor 1000 results from summing over all visible sites. With zero weight, the choice for the initial hidden layer bias is free—by choosing to set it to $-\ln(1/7)$, we are setting the target sparsity to approximately half of that of the visible layer with approximately 100 particles of each species as given by (46). Simulations are run for 500 time steps of size $\Delta t = 0.01$. Figure 4(d) shows the relaxation of the distribution to equilibrium [43].

For training, we use Algorithm 1 with learning rate $\lambda = 0.05$ for the weights and $\lambda = 0.8$ for the biases for 10 000 optimization steps. To estimate the wake phase moments, we sample $\tilde{p}(\boldsymbol{H} = 1|\boldsymbol{V})$ for each sample in a batch size of $\eta = 5$, where $\boldsymbol{V}$ is a data vector. To estimate the sleep phase moments, we alternate between sampling $\tilde{p}[\boldsymbol{H}^{(r)} = 1|\boldsymbol{V}^{(r)}]$ and $\tilde{p}[\boldsymbol{V}^{(r)} = 1|\boldsymbol{H}^{(r-1)}]$ for $r = 1, \ldots, 10$ steps, starting from a random configuration $\boldsymbol{V}^{(0)}$. Alternatively, we also found fast convergence using $k = 10$ steps of CD, as well as using PCD. To reduce the noise in the estimates, we use as is common raw probabilities instead of multinomial states for the hidden units when estimating both the wake and sleep phase moments.

As before, we use the online variant (41) of Algorithm 1 where the limits of integration are shifted during training, with window size $\Delta\tau = 10$, and τ is gradually incremented $\tau \to \tau + 1$ every 100 optimization steps. To learn smooth trajectories and avoid jumps in the learned differential equation model, each time step is divided into 10 substeps when solving the differential equations (44) and (45).

We compare the learned trajectories to a simplified MaxEnt model in Figs. 6(a)–6(c). The side length of the cubic finite elements used was 0.1 on all sides, centered at the initial condition, as shown in Fig. 6(d). Figure 6(a) shows the mean number of particles over the first 100 time steps, as in Fig. 4(d). Figure 6(b) transforms these points to the parameters (b_A, b_B, b_C) of a simple MaxEnt model constrained on these lowest-order moments as given by (46). Figure 6(c) shows the learned model (45), where the biases now control both the means and spatial correlations together with the weights. The trajectory no longer resembles a periodic trajectory, having learned to separate close states in Fig. 6(b). Figure 7 shows the learned time evolution functions for the Rössler oscillator over the first 100 time steps.

The agreement between the stochastic simulations and reconstructed observables is shown in Fig. 8(a). At each time point, 100 samples are drawn from the reduced model by running 25 steps of CD sampling, starting from a random

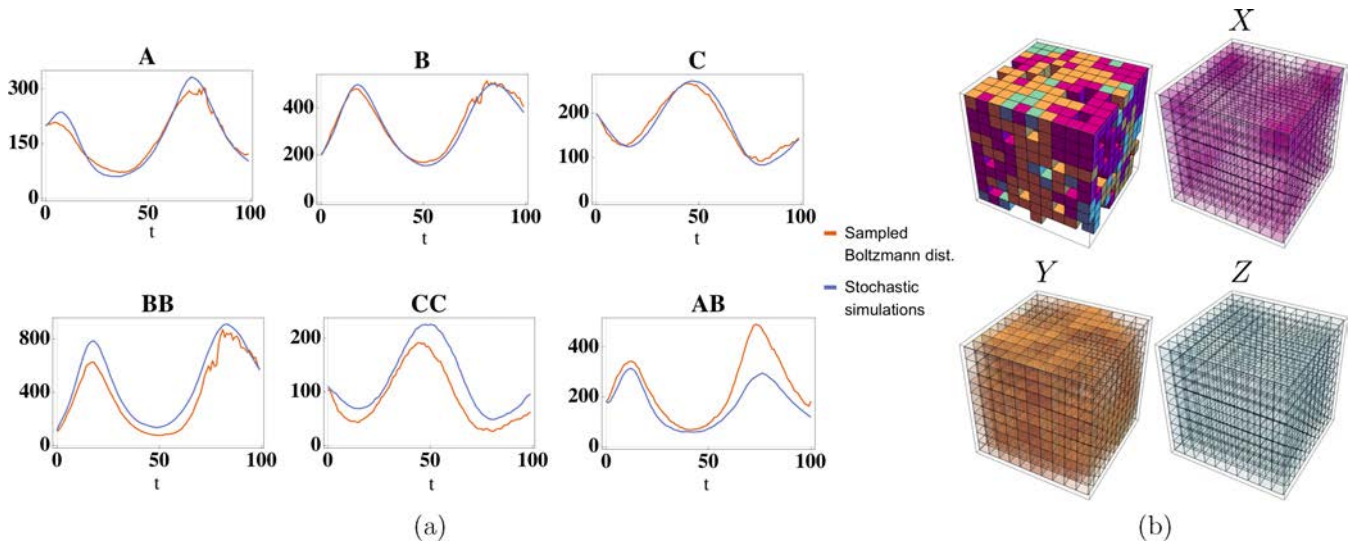

FIG. 8. (a) Example of correlations learned by the reduced model compared to stochastic simulations, obtained by sampling over 100 samples. Top row: Mean number of A, B, C particles. Bottom: Neighboring pairs of (B, B), (C, C), and (A, B). Short range spatial correlations relevant to the moment equations (43) are reasonably approximated due to the chosen connectivity. (a) Sampled state $\boldsymbol{V}$ from the learned model (top left), and the activated hidden layer probabilities $\tilde{p}(\boldsymbol{H}|\boldsymbol{V})$ at time point 20. After training, the hidden layers coarse grain nearest neighbors in the visible layer.

configuration. Nearest neighbors, which determine the time evolution of the means in (43), are reasonably approximated, primarily due to the connectivity chosen in Fig. 5.

Figure 8(b) shows a sampled state $\boldsymbol{V}$ from the learned model, and the activated hidden layer probabilities $\tilde{p}(\boldsymbol{H}|\boldsymbol{V})$ at time point 20. With the learned parameters, the hidden units coarse grain nearest neighbors in the lattice, as needed to approximate the right-hand side of (43). A deeper network such as a deep Boltzmann machine (DBM) may approximate yet-higher spatial correlations and can therefore be used to close differential equation systems depending on higher-order moments.

V. DISCUSSION

We have presented a learning problem for spatiotemporal distributions that estimates differential equation systems controlling a time-varying Boltzmann distribution. The ability to estimate a reduced physical model makes the method interesting for many modeling applications, including chemical kinetics as presented here. Mapping to a differential equation model can likewise be useful for engineering applications, allowing constraints to be efficiently introduced into BM learning as discussed in Sec. III A.

The moment closure approximation presented in Sec. II is broadly applicable due to the use of latent variables that can be trained to capture relevant higher-order correlations rather than deciding *a priori* what correlations to include as in typical closure schemes. Minimizing the KL divergence between the reduced and true models at all times is closely related to entropic matching but differs by the introduction of a differential equation system. We also make the connection to spatially continuous reaction systems explicit.

The finite-element parametrization is similar to the unsupervised learning setting of RBMs in the sense that it is independent of the system under consideration. For deeper architectures such as DBMs as discussed in Sec. IV B, recycling the same time-evolution functions across multiple layers may be effective, similarly to convolution layers in convolutional neural networks. Factoring weights has also been used effectively in deep learning [44] and may similarly reduce the computational burden here. The main advantage of the current DE formalism, however, is to use a parametrization (26) that enforces a physically relevant model.

We have illustrated the advantage of using latent variables in the learning problem, as opposed to a fully visible model. In the fully visible model of Sec. IV A, two and three particle correlations are explicitly captured. In the competing hidden layer model, we use a locally connected RBM (as opposed to fully connected layers) to control the range of correlations captured through the connectivity of the hidden layer. This has the advantage that the representation learned by the hidden layers is easily interpretable as it coarse grains the visible layer. Further, the local connectivity used can be inferred from the moment equations derived from the CME. Deeper networks with multiple hidden layers can be constructed in this fashion to learn hierarchical statistics, with the ability to infer long-range spatial correlations that may become relevant over long timescales.

A popular alternative class of generative models to RBMs are variational autoencoders (VAEs). An adaptation of the proposed method may be possible for these models; however, the main advantages of the current RBM framework is that the form of the energy function can be used interpret the reduced model [22] and that the distribution over the latent variables is not chosen as in VAEs (typically a standard normal distribution) but rather learned from data.

A closely related problem to model reduction is the problem of data assimilation, where noisy measurements and an incomplete model for the dynamics are combined to estimate the true state of the system and unknown parameters in the model [45]. Model reduction methods complement the data

assimilation problem by replacing the physical model with a reduced one which can increase the efficiency of data assimilation methods.

We view the present work as progress toward linking models across scales in biology [20]. Reaction-diffusion systems illustrate many of the common problems in this field. While much machinery (CME or field-theoretic methods) exist to formulate problems for observables, their solution is nontrivial in most applications. Even without analytic challenges such as moment closure, the numerical solution of PDE systems is difficult for systems with high spatial organization or where interactions with other scales (e.g., molecular dynamics) or physics (e.g., electrodiffusion) become relevant. Learning reduced models in the form of spatial dynamic Boltzmann distributions may abstract many of these nontrivial interactions.

ACKNOWLEDGMENTS

This work was supported by NIH R56-AG059602 (E.M., O.K.E., T.M.B., and T.J.S.), Human Frontiers Science Program Grant No. HFSP-RGP0023/2018 (E.M.), NIH P41-GM103712, NIH R01MH115456, AFOSR MURI FA9550-18-1-0051, and NSF DBI-1707356 (O.K.E., T.M.B., and T.J.S.), and the Swartz Foundation (O.K.E. and T.J.S.).

APPENDIX A: FORMAL SOLUTION FOR THE ADJOINT SYSTEM

The connection between (6) and (18) can be made more explicitly. A differential equation system for the perturbations $\delta v_k(\boldsymbol{\alpha}_{\langle i\rangle_k^n}, \boldsymbol{x}_{\langle i\rangle_k^n}, t)$ in (6) can be derived by linearizing the differential equation around a particular solution [22,30]. For the autonomous system (17), this leads to the linear ODE system:

$$\frac{d}{dt}\delta\boldsymbol{v}(\boldsymbol{\alpha},\boldsymbol{x},t) = \delta\boldsymbol{F}(\boldsymbol{\alpha},\boldsymbol{x},t) + G(\boldsymbol{\alpha},\boldsymbol{x},t)\delta\boldsymbol{v}(\boldsymbol{\alpha},\boldsymbol{x},t), \quad \text{(A1)}$$

with some given initial condition $\delta\boldsymbol{v}(\boldsymbol{\alpha},\boldsymbol{x},t_0) = \delta\boldsymbol{\eta}(\boldsymbol{\alpha},\boldsymbol{x})$. Here we have used the vector notation introduced in Sec. II B.

Let the homogenous part of this system

$$\frac{d}{dt}\delta\boldsymbol{v}(\boldsymbol{\alpha},\boldsymbol{x},t) = G(\boldsymbol{\alpha},\boldsymbol{x},t)\delta\boldsymbol{v}(\boldsymbol{\alpha},\boldsymbol{x},t) \quad \text{(A2)}$$

have solution given by the nonsingular fundamental matrix $A(\boldsymbol{\alpha},\boldsymbol{x},t)$. Then (A1) has as formal solution

$$\delta\boldsymbol{v}(\boldsymbol{\alpha},\boldsymbol{x},t) = A(\boldsymbol{\alpha},\boldsymbol{x},t)\left[\delta\boldsymbol{\eta}(\boldsymbol{\alpha},\boldsymbol{x}) + \int_{t_0}^{t} dt'\, A^{-1}(\boldsymbol{\alpha},\boldsymbol{x},t')\delta\boldsymbol{F}(\boldsymbol{\alpha},\boldsymbol{x},t')\right], \quad \text{(A3)}$$

which substituted into (6) gives:

$$\delta S = \int_{t_0}^{t_f} dt \sum_{n=0}^{\infty}\sum_{\boldsymbol{\alpha}}\int d\boldsymbol{x}\; \Delta\boldsymbol{\mu}^{\intercal}(\boldsymbol{\alpha},\boldsymbol{x},t)A(\boldsymbol{\alpha},\boldsymbol{x},t) \times\left[\delta\boldsymbol{\eta}(\boldsymbol{\alpha},\boldsymbol{x}) + \int_{t_0}^{t} dt'\, A^{-1}(\boldsymbol{\alpha},\boldsymbol{x},t')\delta\boldsymbol{F}(\boldsymbol{\alpha},\boldsymbol{x},t')\right] = 0, \quad \text{(A4)}$$

where $\Delta\boldsymbol{\mu}^{\intercal}(t)$ is the vector with components (7). Applying integration by parts on the term in parentheses to move the integral over time gives

$$\left[\delta\boldsymbol{\eta}(\boldsymbol{\alpha},\boldsymbol{x}) + \int_{t_0}^{t_f} dt'\, A^{-1}(\boldsymbol{\alpha},\boldsymbol{x},t')\delta\boldsymbol{F}(\boldsymbol{\alpha},\boldsymbol{x},t')\right] \times\left[\int_{t_0}^{t} dt'\; \Delta\boldsymbol{\mu}^{\intercal}(\boldsymbol{\alpha},\boldsymbol{x},t')A(\boldsymbol{\alpha},\boldsymbol{x},t')\right]\Bigg|_{t=t_0}^{t_f} - \int_{t_0}^{t_f} dt \int_{t_0}^{t} dt'\; \Delta\boldsymbol{\mu}^{\intercal}(\boldsymbol{\alpha},\boldsymbol{x},t')A(\boldsymbol{\alpha},\boldsymbol{x},t') \times A^{-1}(\boldsymbol{\alpha},\boldsymbol{x},t)\delta\boldsymbol{F}(\boldsymbol{\alpha},\boldsymbol{x},t), \quad \text{(A5)}$$

where the adjoint functions $\boldsymbol{\zeta}(t)$ can be identified as:

$$\boldsymbol{\zeta}^{\intercal}(\boldsymbol{\alpha},\boldsymbol{x},t) = \int_{t_0}^{t} dt'\; \Delta\boldsymbol{\mu}^{\intercal}(\boldsymbol{\alpha},\boldsymbol{x},t')A(\boldsymbol{\alpha},\boldsymbol{x},t')A^{-1}(\boldsymbol{\alpha},\boldsymbol{x},t). \quad \text{(A6)}$$

By choosing the adjoint functions to satisfy the boundary condition $\boldsymbol{\zeta}(\boldsymbol{\alpha},\boldsymbol{x},t_f) = 0$, the boundary term in (A5) vanishes and we obtain the previous result (16).

APPENDIX B: DERIVATION OF MOMENT EQUATIONS FROM THE CHEMICAL MASTER EQUATION

The moment equations (36) and (43) can be derived from the chemical master equation using the Doi-Peliti [46] formalism and its equivalent generating function representation. We demonstrate this for the Rössler system (43).

For notational convenience, we do not consider the single-occupancy limit here. The state of the system is described by the $N \times M$ matrix $\boldsymbol{V}'$ with entries $v_{i,\alpha} \in \{0, 1, 2, \dots\}$, where $N = 10 \times 10 \times 10$ rows denote lattice sites, and $M = 3$ columns denote occupancies of species $\{A, B, C\}$.

Define the $N \times M$ single-entry *matrix* $\boldsymbol{e}_{ij}$ with entries zero everywhere except at index (i, j) where it is one. The creation and annihilation operators $\hat{a}_{i,\alpha}$ and $a_{i,\alpha}$ create and destroy particles of species α at unit i:

$$\hat{a}_{i,\alpha}|\boldsymbol{V}'\rangle = |\boldsymbol{V}' + \boldsymbol{e}_{i,\alpha}\rangle, \qquad a_{i,\alpha}|\boldsymbol{V}'\rangle = v_{i,\alpha}|\boldsymbol{V}' - \boldsymbol{e}_{i,\alpha}\rangle. \quad \text{(B1)}$$

The operators corresponding to reactions in the Rössler system (excluding diffusion) are then:

$$A \to 2A: \; k_1 \sum_{i=1}^{N}(\hat{a}_{i,A} - 1)\hat{a}_{i,A}a_{i,A},$$

$$2A \to A: \; \kappa_1 \sum_{\langle ij\rangle}(1 - \hat{a}_{j,A})\hat{a}_{i,A}a_{i,A}a_{j,A},$$

$$A + B \to 2B: \; \kappa_2 \sum_{\langle\langle ij\rangle\rangle}(\hat{a}_{i,B} - \hat{a}_{i,A})\hat{a}_{j,B}a_{i,A}a_{j,B},$$

$$A + C \to \varnothing: \; \kappa_3 \sum_{\langle\langle ij\rangle\rangle}(1 - \hat{a}_{i,A}\hat{a}_{j,C})a_{i,A}a_{j,C},$$

$$B \to \varnothing: \; k_2 \sum_{i=1}^{N}(1 - \hat{a}_{i,B})a_{i,B},$$

$$C \to 2C: \; k_3 \sum_{i=1}^{N} (\hat{a}_{i,C} - 1)\hat{a}_{i,C} a_{i,C},$$

$$2C \to C: \; \kappa_4 \sum_{\langle ij \rangle} (1 - \hat{a}_{j,C})\hat{a}_{i,C} a_{i,C} a_{j,C}, \tag{B2}$$

where $\sum_{\langle ij \rangle}$ sums over all neighboring sites without double counting, $\sum_{\langle\langle ij \rangle\rangle}$ sums over all neighboring sites with double counting, and we specify the species $\{A, B, C\}$ instead of an index $\alpha = 1, \ldots, M$ for clarity in the subscripts. Here we place new particles resulting from fission reactions with rates k_1 and k_3 at the same site - in the single-occupancy limit, they must be placed at a neighboring site. For bimolecular reactions with rates κ_1 and κ_4, we make the in this case ambiguous choice to place new species at site i versus j. The time evolution operator W for the Rössler system is the sum of all terms in (B2).

The system state and the ladder operators admit an equivalent generating function representation:

$$|\boldsymbol{V}'\rangle \to \prod_{i=1}^{N} \prod_{\alpha=1}^{M} z_{i,\alpha}^{v_{i,\alpha}}, \quad \hat{a}_{i,\alpha} \to z_{i,\alpha}, \quad a_{i,\alpha} \to \frac{\partial}{\partial z_{i,\alpha}}. \tag{B3}$$

An observable $\langle X \rangle$ with generating function representation X_z according to (B3) evolves as:

$$\frac{d\langle X \rangle}{dt} = \left(X_z W \prod_{i=1}^{N} \prod_{\alpha=1}^{M} z_{i,\alpha}^{v_{i,\alpha}} \right) \Big|_{z=1}, \tag{B4}$$

where W is now the sum of terms (B2) in the generating function representation (B3). From the number operator $\hat{a}_{k,\beta} a_{k,\beta}$ which counts the number of particles of species β at position k, the time evolution of the mean number of particles of species β is then

$$\frac{d\mu_\beta}{dt} = \left(\sum_{k=1}^{N} z_{k,\beta} \frac{\partial}{\partial z_{k,\beta}} W \prod_{i=1}^{N} \prod_{\alpha=1}^{M} z_{i,\alpha}^{v_{i,\alpha}} \right) \Big|_{z=1}, \tag{B5}$$

which can be directly evaluated to give the moment equations (43). For a review on field theoretic methods for reaction-diffusion systems, we refer to Mattis and Glasser [47]. The formalism can also describe systems in continuous space [46] where it has a similar generation function representation [22].

[1] S. Hellander, A. Hellander, and L. Petzold, Phys. Rev. E **91**, 023312 (2015).
[2] T. Ramalho, M. Selig, U. Gerland, and T. A. Enßlin, Phys. Rev. E **87**, 022719 (2013).
[3] A. Ruttor and M. Opper, Phys. Rev. Lett. **103**, 230601 (2009).
[4] L. Bronstein and H. Koeppl, Phys. Rev. E **97**, 062147 (2018).
[5] O. Marre, S. El Boustani, Y. Frégnac, and A. Destexhe, Phys. Rev. Lett. **102**, 138101 (2009).
[6] C. O'Donnell, J. T. Gonçalves, N. Whiteley, C. Portera-Cailliau, and T. J. Sejnowski, Neur. Comput. **29**, 50 (2016).
[7] K. Zhao, T. Osogami, and R. Raymond, in *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA* (NIPS, 2017).
[8] Y. Li, R. Yu, C. Shahabi, and Y. Liu, *International Conference on Learning Representations* (ICLR, Vancouver, Canada, 2018).
[9] S. Lefèvre, D. Vasquez, and C. Laugier, Robomech. J. **1**, 1 (2014).
[10] C. W. Gardiner, *Stochastic Methods: A Handbook for the Natural and Social Sciences* (Springer, Berlin, 2009).
[11] B. Munsky and M. Khammash, J. Chem. Phys. **124**, 044104 (2006).
[12] G. Uhlenbeck and G. Ford, *Lectures in Statistical Mechanics*, Lectures in Applied Mathematics (American Mathematical Society, Washington, DC, 1963).
[13] L. Bronstein and H. Koeppl, J. Chem. Phys. **148**, 014105 (2018).
[14] P. Smadbeck and Y. N. Kaznessis, Proc. Natl. Acad. Sci. USA **110**, 14261 (2013).
[15] T. Johnson, T. Bartol, T. Sejnowski, and E. Mjolsness, Phys. Biol. **12**, 045005 (2015).
[16] D. T. Gillespie, J. Phys. Chem. **81**, 2340 (1977).
[17] J. Stiles and T. Bartol, in *Computational Neuroscience* (CRC Press, Boca Raton, FL, 2000).
[18] R. A. Kerr, T. M. Bartol, B. Kaminsky, M. Dittrich, J.-C. J. Chang, S. B. Baden, T. Sejnowski, and J. R. Stiles, SIAM J. Sci. Comput. **30**, 3126 (2008).
[19] P. Thomas, R. Grima, and A. V. Straube, Phys. Rev. E **86**, 041110 (2012).
[20] E. Mjolsness, Bull. Math. Biol. (2019), doi: 10.1007/s11538-019-00628-7.
[21] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, Cogn. Sci. **9**, 147 (1985).
[22] O. K. Ernst, T. Bartol, T. Sejnowski, and E. Mjolsness, J. Chem. Phys. **149**, 034107 (2018).
[23] P. Mehta, M. Bukov, C.-H. Wang, A. G. Day, C. Richardson, C. K. Fischer, and D. J. Schwab, Phys. Rep. **810**, 1 (2019).
[24] Y. Bengio, A. Courville, and P. Vincent, IEEE Trans. Pattern Anal. Mach. Intell. **35**, 1798 (2013).
[25] D. MacKay, *Information Theory, Inference, and Learning Algorithms* (Cambridge University Press, Cambridge, 2003).
[26] G. Carleo and M. Troyer, Science **355**, 602 (2017).
[27] Z.-Y. Han, J. Wang, H. Fan, L. Wang, and P. Zhang, Phys. Rev. X **8**, 031012 (2018).
[28] R. V. Gamkrelidze and G. L. Kharatishvili, Math. Syst. Theory **1**, 229 (1967).
[29] L. W. Neustadt, in *Symposium on Optimization*, edited by A. V. Balakrishnan, M. Contensou, B. F. de Veubeke, P. Krée, J. L. Lions, and N. N. Moiseev (Springer, Berlin, 1970), pp. 292–306.
[30] R. V. Gamkrelidze, *Principles of Optimal Control Theory* (Springer US, Boston, MA, 1978).
[31] Y. Cao, S. Li, L. Petzold, and R. Serban, SIAM J. Sci. Comput. **24**, 1076 (2003).
[32] P. Smolensky, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol. 1, edited by D. E. Rumelhart, J. L. McClelland and CORPORATE PDP Research Group (MIT Press, Cambridge, MA, USA, 1986), pp. 194–281.

[33] Z. Ghahramani, in *Adaptive Processing of Sequences and Data Structures* (Springer, Berlin, 1998), pp. 168–197.
[34] D. P. Kingma and J. Ba, arXiv:1412.6980.
[35] T. Tieleman, in *Proceedings of the 25th International Conference on Machine Learning* (ACM, New York, 2008), pp. 1064–1071.
[36] R. Herzog and K. Kunisch, GAMM-Mitteilungen **33**, 163 (2010).
[37] S. W. Funke and P. E. Farrell, arXiv:1302.3894.
[38] T. Hughes, *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis* (Dover, Mineola, NY, 2000).
[39] D. Arnold and A. Logg, SIAM News **47**, 1 (2014).
[40] H. Takayasu and A. Y. Tretyakov, Phys. Rev. Lett. **68**, 3060 (1992).
[41] W. K. D. and O. E. Rössler, Zeitschr. Naturforsch. A **35**, 317 (1980).
[42] G. Bellesia and B. B. Bales, Phys. Rev. E **94**, 042306 (2016).
[43] V. Anishchenko, T. Vadivasova, G. Strelkova, and G. Okrokvertskhov, Math. Biosci. Eng. **1**, 161 (2004).
[44] M. Ranzato, A. Krizhevsky, and G. Hinton, in *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, (AISTATS) 2010, Sardinia, Italy*, edited by Y. W. Teh and M. Titterington, Proceedings of Machine Learning Research (PMLR, 2010), Vol. 9, pp. 621–628.
[45] H. D. I. Abarbanel, *Predicting the Future: Completing Models of Observed Complex Systems* (Springer, New York, 2013).
[46] M. Doi, J. Phys. A: Math. Gen. **9**, 1465 (1976).
[47] D. C. Mattis and M. L. Glasser, Rev. Mod. Phys. **70**, 979 (1998).