10-1-2019

Model reconstruction from temporal data for coupled oscillator networks

Mark J. Panaggio
Hillsdale College

Maria Veronica Ciocanel
The Ohio State University

Lauren Lazarus
Trinity College Hartford

Chad M. Topaz
Williams College

Bin Xu
University of Notre Dame

Trinity College
HARTFORD CONNECTICUT

Model reconstruction from temporal data for coupled oscillator networks

Mark J. Panaggio, Maria-Veronica Ciocanel, Lauren Lazarus, Chad M. Topaz, and Bin Xu

COLLECTIONS

Paper published as part of the special topic on When Machine Learning Meets Complex Systems: Networks, Chaos and Nonlinear Dynamics

View Online   Export Citation   CrossMark

ARTICLES YOU MAY BE INTERESTED IN
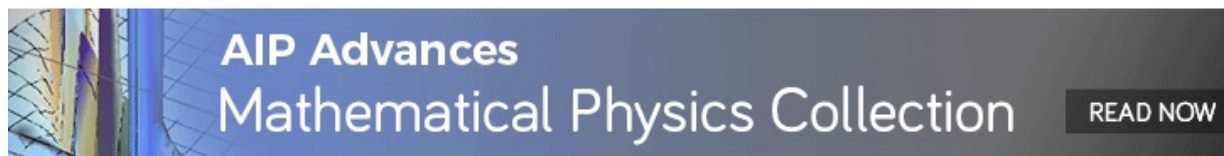
Reconstructing dynamical networks via feature ranking
Chaos: An Interdisciplinary Journal of Nonlinear Science **29**, 093107 (2019); https://doi.org/10.1063/1.5092170

Good and bad predictions: Assessing and improving the replication of chaotic attractors by means of reservoir computing
Chaos: An Interdisciplinary Journal of Nonlinear Science **29**, 103143 (2019); https://doi.org/10.1063/1.5118725

Introduction to Focus Issue: When machine learning meets complex systems: Networks, chaos, and nonlinear dynamics
Chaos: An Interdisciplinary Journal of Nonlinear Science **30**, 063151 (2020); https://doi.org/10.1063/5.0016505

AIP Advances
Mathematical Physics Collection
READ NOW

Model reconstruction from temporal data for coupled oscillator networks

View Online　　Export Citation　　CrossMark

Mark J. Panaggio,[1,a)] (iD) Maria-Veronica Ciocanel,[2] (iD) Lauren Lazarus,[3] (iD) Chad M. Topaz,[4] (iD) and Bin Xu[5]

AFFILIATIONS

[1] Department of Mathematics, Hillsdale College, Hillsdale, Michigan 49242, USA
[2] Mathematical Biosciences Institute, The Ohio State University, Columbus, Ohio 43210, USA
[3] Department of Mathematics, Trinity College, Hartford, Connecticut 06106, USA
[4] Department of Mathematics and Statistics, Williams College, Williamstown, Massachusetts 01267, USA
[5] Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, Indiana 46556, USA

Note: This paper is part of the Focus Issue, "When Machine Learning Meets Complex Systems: Networks, Chaos and Nonlinear Dynamics."
[a)] **Electronic mail:** mpanaggio@hillsdale.edu

ABSTRACT

In a complex system, the interactions between individual agents often lead to emergent collective behavior such as spontaneous synchronization, swarming, and pattern formation. Beyond the intrinsic properties of the agents, the topology of the network of interactions can have a dramatic influence over the dynamics. In many studies, researchers start with a specific model for both the intrinsic dynamics of each agent and the interaction network and attempt to learn about the dynamics of the model. Here, we consider the inverse problem: given data from a system, can one learn about the model and the underlying network? We investigate arbitrary networks of coupled phase oscillators that can exhibit both synchronous and asynchronous dynamics. We demonstrate that, given sufficient observational data on the transient evolution of each oscillator, machine learning can reconstruct the interaction network and identify the intrinsic dynamics.

Many complex systems in nature, science, and society can be modeled as a network of coupled oscillators. The interactions between neurons in the brain, generators in a power grid, and individuals within a crowd or social network can cause unexpected behaviors to emerge. Often, it is possible to observe the behavior of individual oscillators, but not to directly determine their intrinsic properties or the structural connections between them. In this paper, we consider the problem of reconstructing the structural interaction network, the functional form of the coupling, and the oscillator dynamics from time series data. We propose a method for computing the optimal reconstruction using techniques commonly used for training artificial neural networks. For systems that synchronize, we explore various minimally invasive types of perturbations that can be used to temporarily disrupt the synchronization, generating additional useful transient data for reconstruction. Our method can accurately reconstruct the model for systems of phase oscillators over a wide range of parameters and dynamic regimes using less data than alternative methods.

I. INTRODUCTION

Nature and society brim with systems of coupled oscillators, including pacemaker cells in the heart, insulin-secreting cells in the pancreas, networks of neurons in the brain, cardiorespiratory synchronization, fireflies that synchronize their flashing, chemical reactions, Josephson junctions, power grids, metronomes, and applause in human crowds, to name merely a few.[1–10] The dynamics of coupled oscillators in complex networks have been studied extensively. In particular, networks of interacting oscillators governed by the seminal Kuramoto model[11–13] or its variants are known to exhibit a rich variety of behaviors including spontaneous synchronization, phase transitions, and pattern formation.[14,15] The connections in the network can play a pivotal role in determining these dynamics.

When the governing equations and network topology for a system of coupled oscillators are known, it is fairly straightforward to explore the dynamics of the system using numerical methods and, in special cases, analytical methods. Unfortunately, in many systems, the topology of the interaction network and the intrinsic oscillator properties can be difficult or impossible to observe directly. Imagine, for example, experimental results that track neuronal cell network activity or gene regulatory network activity for a large number of oscillators, giving rise to time series data. One might like to infer models and connectivity information from these time series.

It is crucial to distinguish between functional network connections and structural network connections. Functional connectivity refers to the temporal correlation of the oscillators and is often directly observable. For instance, neurons that fire synchronously are functionally connected. In contrast, structural connectivity, also called network topology, refers to the underlying connections present in the network. For instance, neurons that interact via synapses or neurotransmitters are structurally connected. Unfortunately, these structural connections may not be observable without deconstructing the network.

Network reconstruction is an active area of research, both in theoretical work focused on observations from complex physical systems[16–30] and in applied fields where simultaneously-recorded physiological signals are used to reconstruct physiological networks of interaction.[31–33] Our primary contribution is a new algorithm for solving this inverse problem. We use observed time series data consisting of the phases from a network of coupled oscillators to reconstruct both the network topology, i.e., the structural network connections and the intrinsic oscillator dynamics. For the remainder of this paper, when we refer to inferring or reconstructing the model, we mean both of the aforementioned components: network connections and intrinsic oscillator properties.

In Sec. II B, we review the literature most closely related to our proposed model reconstruction technique. This technique addresses certain challenges of existing methods. One such challenge is that existing methods set up the inverse problem as a linear system that involves many unknown parameters. Consequently, a large amount of time series data is necessary. In contrast, our algorithm results in a system of equations that is smaller and is nonlinear. We solve these using optimization tools designed for training artificial neural networks. As a result, we are able to infer the model with a much smaller amount of data. A second challenge, as mentioned in Refs. 18, 22, 28, and 29, is that it is difficult to infer the model for networks that are synchronized. For such networks, we explore which perturbations of the synchronized state are needed to enable accurate inference of the model.

The mathematical setting of our study is oscillator networks composed of limit-cycle oscillators. These are also known as phase oscillators, meaning they are characterized by a single phase variable defined on the circle. We explore four different choices of oscillator models, all of which fall into the general framework for uniformly coupled phase oscillators developed in Ref. 34: the classic Kuramoto model,[11] a Kuramoto-like model with square-wave coupling function, the Kuramoto-Sakaguchi model,[35] and a phase-field reduction of weakly-coupled Hodgkin-Huxley oscillators[36] (see Appendix A). We take all networks to be Erdös-Rényi random graphs.[37] We attempt to infer the network topology, the intrinsic frequency of each model

oscillator, and the so-called coupling function that specifies the influence of one oscillator on others.

To carry out the model reconstruction, we begin with simulated time series for the oscillator phases and estimated phase velocities. We set up an optimization problem involving the mean squared error for the predicted phase velocities generated by a system of nonlinear differential equations, which can be represented as a convolutional neural network. We then use numerical minimization methods to estimate the optimal parameters, thereby inferring the adjacency matrix for network connectivity, the oscillator coupling strength, the oscillator frequencies, and the Fourier coefficients of the coupling function. This approach is effective for a variety of different networks and model parameters. More specifically, we find the following:

- Accurate inference of the coupling network, frequencies, and coupling functions is possible independent of the model when the system does not synchronize or when sufficient perturbations from synchronization are permitted.
- For small data sets, model reconstruction with our method is more accurate than with alternative methods based on solving linear systems of equations.
- Computational methods designed for training artificial neural networks, such as mini-batch gradient descent implemented in TensorFlow,[38] can be used to solve the optimization problem associated with model reconstruction.
- Synchronizing networks can be reconstructed using random phase resets or sufficiently large phase perturbations to a small, randomly chosen set of oscillators.
- Synchronizing networks can also be reconstructed using perturbations to a sufficiently large fixed subset of oscillators.

The rest of this paper is organized as follows. In Sec. II, we introduce the Kuramoto model (a standard model for the dynamics of coupled oscillators), set up the inverse problem for reconstruction, and discuss approaches for solving this inverse problem. In Sec. III, we describe a series of experiments used to test the effectiveness of these reconstruction techniques over a wide range of model parameters. In Sec. IV, we present the results of the aforementioned experiments and discuss perturbation strategies for improving the reconstruction when the system synchronizes. Finally, in Sec. V, we summarize our primary findings and discuss extensions to our methodology.

II. MODELS AND METHODS

The Kuramoto model is a standard model for the dynamics of coupled oscillators. We consider N oscillators with phases $\theta_k \in [0, 2\pi)$ for $k = 1, 2, \ldots, N$, each with an intrinsic frequency ω_k. These oscillators are coupled through an interaction network with adjacency matrix A, with the entry $A_{kj} \in \{0, 1\}$ determining whether oscillators k and j are coupled. The dynamics of θ_k are governed by the equation

$$\dot{\theta}_k = \omega_k + \sum_{j=1}^{N} \sigma_{kj} A_{kj} \Gamma(\theta_j - \theta_k). \tag{1}$$

Here, σ_{kj} represents the strength of the coupling between oscillators j and k, and $\Gamma(\theta)$ represents the coupling function. If the coupling function $\Gamma(\theta)$ is continuous and 2π-periodic, and $\Gamma'(\theta)$ is piecewise

TABLE I. Summary of network model parameters, with the default values and value ranges considered in the parameter sweep experiments of Sec. III.

Parameter	Description	Value	Sweep values
N	Number of oscillators	10	$\{5, 10, 20, 40\}$
μ	Average intrinsic frequency	1.0	N/A
σ	Standard deviation of intrinsic frequencies	0.5	$\{0.01, 0.1, 1.0\}$
p	Network connection probability	0.5	$\{0.1, 0.2, \ldots, 0.9\}$
$\Gamma(\theta)$	Coupling function	$\sin(\theta)$	See Appendix A
dyn_noise	Noise in system dynamics	0	$\{0, 10^{-5}, 10^{-4}, \ldots, 10^0\}$

continuous, then it can be represented using a uniformly convergent Fourier series

$$\Gamma(\theta) = a_0 + \sum_{n=1}^{\infty} a_n \cos(n\theta) + b_n \sin(n\theta), \qquad (2)$$

where the coefficients satisfy $a_n, b_n \leq M/n^2$ for some M. Therefore, these coefficients decay to 0 for large n, and this function can be approximated to arbitrary accuracy using a truncated Fourier series

$$\Gamma(\theta) = a_0 + \sum_{n=1}^{m} [a_n \cos(n\theta) + b_n \sin(n\theta)] \qquad (3)$$

with sufficiently large m. Note that within the model, one can set $a_0 = 0$ without loss of generality, using the change of variables $\omega_k + \sum_{j=1}^{N} \sigma_{kj} A_{kj} a_0 \rightarrow \omega_k$.

In Kuramoto's original formulation,[11] the oscillators were globally coupled ($A_{kj} = 1$) with coupling strengths $\sigma_{kj} = K/N$, where K scales the global coupling strength. He used the coupling function $\Gamma(\theta) = \sin(\theta)$ and showed that, for $K > K_C$ where K_C is a critical value related to the width of the distribution of intrinsic frequencies, the oscillators begin to synchronize, achieving identical phase velocities with phases distributed around the population mean. Analogous results were obtained later for systems with periodic coupling functions possessing only odd harmonics[12,34] and arbitrary complex networks.[39]

A. Simulated data generation

We investigate a method for inferring the parameters of the Kuramoto model for phase oscillators on random graphs with uniform coupling strengths $\sigma_{kj} = K/N$. In our experiments, we focus on undirected Erdös-Rényi graphs, where each possible edge in the network is present with a probability of p. It is straightforward to extend our approach to arbitrary networks with nonuniform coupling strengths, though this would likely require a larger amount

of time series data. The details of our method appear in Secs. II B and II C.

In order to test our method, we generate data according to the following procedure:

1. Generate an Erdös-Rényi network for a fixed connection probability p, where the nodes represent individual oscillators with natural frequencies sampled from a normal distribution with mean μ and standard deviation σ.
2. Use numerical integration to generate a time series for the evolution of this system for $t \in [0, t_{\max}]$, starting from initial phases drawn from a uniform distribution on $[0, 2\pi]$. For numerical integration, we use an explicit Runge-Kutta method of order 5(4) (Ref. 40) or a stochastic Runge-Kutta method of order 2 (Ref. 41).
3. Compute the phases $\theta_k(t_n)$ at times $t_n = n\Delta t$ with time step Δt and for $n = 0, 1, \ldots, T + 1$, where $T = t_{\max}/\Delta t - 1$, to obtain observations that are evenly spaced in time.
4. Estimate the phase velocities $v_k(t_n) = \dot{\theta}_k(t_n)$ for $n = 1, 2, \ldots, T$, using central differencing in time with Savitsky-Golay filtering.[42] We use a window length of 5 with first degree polynomials.
5. Repeat steps (2–4) N_{res} times with different uniform random initial conditions to obtain sufficient data during the transient evolution of the system.

See Tables I and II for the network parameters and numerical solution parameters used in this procedure. For these default values, one typically finds that some of the oscillators synchronize with a collective rhythm, while others drift relative to this collective rhythm.

The aforementioned process is intended to produce simulated data mimicking that which experimentalists might collect when observing real world networks. Of course, it may not be possible to control the initial phases in an experiment. We, therefore, evaluate the reconstruction methods for varying N_{res}, as well as for cases in which small perturbations are used instead of different initial conditions (see Sec. IV A).

TABLE II. Summary of algorithm parameters, with the default values and value ranges considered in the parameter sweep experiments of Sec. III.

Parameter	Description	Value	Sweep values
$t_{\max}$	Duration of each transient	20	$\{2, 5, 10, 20, 50\}$
Δt	Sampling time step	0.1	N/A
Noise	Observation noise	0	$\{0, 10^{-5}, 10^{-4}, \ldots, 10^0\}$
N_{res}	Number of transients observed	10	$\{1, 2, 5, 10, 20, 40\}$

B. Inverse problem formulation

We now formulate a set of equations whose least squares solution can be used to estimate the natural frequencies ω_k of each oscillator, the adjacency matrix A_{kj} of the coupling network, the coupling strength K, and the coupling function $\Gamma(\theta)$, from observed phases $\theta_k(t_n)$ and phase velocities $v_k(t_n)$ generated using the method outlined above. This work builds on prior work by Shandilya and Timme,[18] where the adjacency matrix for a network of oscillators is estimated from a time series. We summarize their approach below.

Define vectors $\boldsymbol{\theta}_j = [\theta_1(t_j), \theta_2(t_j), \ldots, \theta_N(t_j)]^T$ and $\boldsymbol{v}_j = [\dot{\theta}_1(t_j), \dot{\theta}_2(t_j), \ldots, \dot{\theta}_N(t_j)]^T$ consisting of the oscillator phases and phase velocities. Note that here ^{T} corresponds to the matrix transpose. When the coupling strength K and coupling function Γ are known, the phase velocity for oscillator k,

$$(\boldsymbol{v}_j)_k = \omega_k + \frac{K}{N} \sum_{j=1}^{N} A_{kj} \Gamma(\theta_j - \theta_k), \qquad (4)$$

is linear in the unknown parameters ω_k and A_{kj}. Therefore, one can infer both the natural frequencies and adjacency matrix by solving a linear system of $N \cdot T$ equations, where T is the number of time steps, and each time step j provides N equations of the form

$$L(\boldsymbol{\theta}_j) \boldsymbol{x} = \boldsymbol{v}_j. \qquad (5)$$

Here, $L(\boldsymbol{\theta}_j)$ is an $N \times (N + N^2)$ matrix where each row is determined from Eq. (4) for a particular oscillator, and $\boldsymbol{x}$ is an $(N + N^2) \times 1$ vector with the unknown natural frequencies $\boldsymbol{\omega}$ and the entries in the adjacency matrix A. The number of unknowns in A can be reduced to $N + N(N-1)/2$ if one assumes $A_{jj} = 0$ and $A_{kj} = A_{jk}$.

Since the number of equations in the linear system is determined by the number of observed time steps T and the number of oscillators N, one can ensure that the system is overdetermined by collecting enough observations so that $T > 1 + N$. One can then estimate the parameters by minimizing a loss function such as the mean squared error,

$$E(\boldsymbol{\omega}, A) = \frac{1}{NT} \sum_{j=1}^{T} \|\boldsymbol{v}_j - \hat{\boldsymbol{v}}_j(\boldsymbol{\omega}, A)\|_2^2,$$

where $\boldsymbol{v}_j$ denotes the observed value and $\hat{\boldsymbol{v}}_j$ denotes the predicted value of the velocity.

As long as the system remains far from synchronization, increasing the number of observations T will provide additional linearly independent equations to aid with reconstruction. However, as networks synchronize, i.e., $(\boldsymbol{v}_j)_k = \dot{\theta}_k \to \dot{\theta}$, additional observations become (nearly) linearly dependent. This causes the system of equations to become highly ill-conditioned and makes numerical solutions sensitive to noise and rounding errors. As a result, it is often necessary to perturb the system away from equilibrium to ensure that a sufficient number of linearly independent observations can be collected.

Although the approach of Ref. 18 outlined above is effective, it is limited by the requirement that the coupling function be known *a priori*. This limitation was addressed in the context of pulse-coupled oscillators in Ref. 28 and phase-coupled oscillators in Ref. 29 by representing the unknown functions as a Fourier series so that it can

also be estimated by solving an optimization problem. Unfortunately, one challenge remains. If one represents the coupling function as in Eq. (3), then the system of equations is no longer linear as terms of the form $A_{kj}a_n$ and $A_{kj}b_n$ appear. Pikovsky[29] circumvents this issue by defining distinct coupling functions

$$\Gamma_{kj}(\theta) = \sum_{n=1}^{m} a_{kjn} \cos(n\theta) + b_{kjn} \sin(n\theta)$$

for each pair of oscillators and by setting $A_{kj} = 1$. In this formulation, if oscillators k and j are uncoupled, then $a_{kjn} = b_{kjn} = 0$ for all n. This modification preserves the linearity of the system of equations and allows for the description of a more general class of networks with distinct coupling functions for each pair of oscillators. However, it comes at a cost: the number of unknown parameters increases dramatically from $N + N^2$ to $N + 2mN^2$. Therefore, even longer observation times are required. Additionally, as with the method of Ref. 18, if the system synchronizes, there may be numerical difficulties when inferring parameters. Finally, the large number of free parameters makes this model particularly prone to overfitting. As before, one could assume that connections in the network are symmetric, i.e., that $\Gamma_{kj}(\theta) = \Gamma_{jk}(\theta)$ and that self edges are not included $\Gamma_{jj}(\theta) = 0$, allowing for a reduction in the number of parameters to $N + mN(N-1)$.

We propose an alternative approach. We use a single coupling function represented by a Fourier series as described in Eq. (3) for all coupling terms and attempt to infer the $2m$ Fourier coefficients $\boldsymbol{a} = [a_1, a_2, \ldots, a_m]^T$ and $\boldsymbol{b} = [b_1, b_2, \ldots, b_m]^T$ along with the intrinsic frequencies $\boldsymbol{\omega}$, adjacency matrix A, and global coupling strength K. This leads to a total $2m + N + N^2 + 1$ inferred parameters or $2m + N + N(N-1)/2 + 1$ with symmetry constraints and no self-connections. This is, therefore, a modest increase of $2m + 1$ parameters over the case considered by Shandilya and Timme[18] without requiring prior knowledge of the coupling function.

As mentioned previously, this leads to a set of nonlinear equations due to the appearance of terms of the form $KA_{kj}a_n$ and $KA_{kj}b_n$ in Eq. (4). Since K always appears in products involving a_n and b_n, these parameters are not structurally identifiable. One could set $K = 1$ without loss of generality. Instead, we include K as an inferred parameter and introduce penalty terms to the objective function we seek to minimize,

$$\begin{aligned}
E(\boldsymbol{\omega}, A, \mathbf{a}, \mathbf{b}) &= \frac{1}{NT} \sum_{j=1}^{T} \|\boldsymbol{v}_j - \hat{\boldsymbol{v}}_j(\boldsymbol{\omega}, A, \boldsymbol{a}, \boldsymbol{b}, K)\|_2^2 \\
&\quad + \lambda_\Gamma \sum_{n=1}^{\infty} \left(|a_n|^2 + |b_n|^2 \right) \\
&\quad + \lambda_A \sum_{j=1}^{N} \sum_{k=1}^{N} \left(|A_{kj}|^2 + |1 - A_{kj}|^2 \right) \\
&\quad + \lambda_{bd} \sum_{j=1}^{N} \sum_{k=1}^{N} \left(\min\{A_{kj}, 0\} + \min\{1 - A_{kj}, 0\} \right).
\end{aligned}$$

$$(6)$$

TABLE III. Default values and summary of method parameters for results in Sec. IV.

Parameter	Description	Value
n_epochs	Iterations through the training data	300
Batch size	Time steps of data per gradient descent batch	100
m	Number of inferred Fourier coefficients	5
λ_Γ	Weight of penalty on nonsparse coupling	0.0001
λ_A	Weight of penalty on network connectivity	10^{-6}
λ_{bd}	Weight of penalty on $A_{kj} \notin [0,1]$	10^5

The inclusion of a penalty function ensures the existence of a nondegenerate local minimum. In this objective function, the first term represents the mean squared error. The remaining terms are penalty terms with weights $\lambda_\Gamma = 0.0001$, $\lambda_A = 10^{-6}$, and $\lambda_{bd} = 10^5$. These hyperparameter values were tuned initially to provide satisfactory reconstruction performance and then kept constant in all subsequent experiments (see also Table III). Additional tuning of these parameters for specific networks could further improve the accuracy of the model reconstruction. The term with λ_Γ introduces L_2 regularization, which favors smaller estimates of the parameter values a_n and b_n. This is useful for combating overfitting and promoting sparse representations and is analogous to using a Bayesian prior centered at 0 for the Fourier coefficients.[43] The term with λ_A penalizes values in the adjacency matrix that are not close to 0 or 1. The last term with λ_{bd} penalizes adjacency values that are negative or greater than 1 to ensure that the estimates fall within the desired range of $[0,1]$. We do not penalize K and instead allow it to be as large as necessary to counteract the parameter shrinkage caused by the penalty terms.

C. Inverse problem solution

Minimizing Eq. (6) poses a number of challenges. First of all, due to the nonlinearity of Eq. (4), this function may not be convex, and there are no guarantees that local optimization methods will converge to a global minimum. Secondly, when the number of observations T is large, this function may be costly to compute, and minimizing the number of function evaluations is paramount.

Fortunately, the theoretical challenge of nonconvexity does not prevent our method from obtaining consistently reliable model reconstruction, as we show in Sec. IV. The computational difficulties can be addressed by using tools designed for artificial neural networks and implemented in TensorFlow to efficiently compute gradients and perform minimization with a variant of mini-batch gradient descent.[44]

To elucidate the connection with artificial neural networks, we comment that Eq. (3) can be viewed as a two-dimensional convolutional neural network with a convolution of size 1×1 and stride one applied to a tensor of phase differences $\Delta\theta_{i,j,k} = \theta_k(t_i) - \theta_j(t_i)$ with $i = 1, 2, \ldots T$ and $k, j = 1, 2, \ldots N$. This network consists of an input layer with dimensions $T \times N \times N$ (the dimensions of the phase difference tensor), a hidden layer with dimensions $T \times N \times N \times 2m$ corresponding to each of the $2m$ harmonics of the form $\cos n\Delta\theta_{i,j,k}$ and $\sin n\Delta\theta_{i,j,k}$, followed by a $T \times N \times N$ output layer representing $\Gamma(\Delta\theta_{i,j,k}) = \sum_{n=1}^{m} \left[a_n \cos n(\Delta\theta_{i,j,k}) + b_n \sin n(\Delta\theta_{i,j,k}) \right]$. A traditional
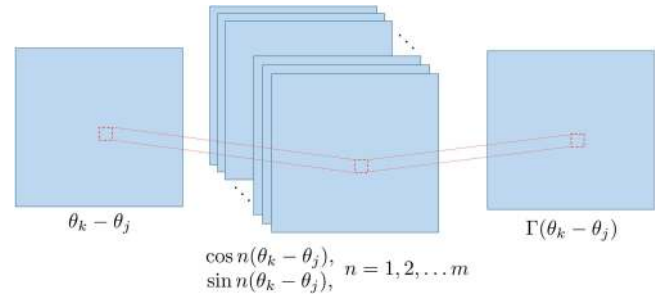

FIG. 1. Coupling function as a convolutional neural network. This schematic illustrates how the coupling function Γ [Eq. (3)] is evaluated at each phase difference $\theta_k - \theta_j$ via a convolutional neural network with size 1×1 and stride one. The hidden layer consists of $2m$ units with activation functions $\cos(n\theta)$ and $\sin(n\theta)$, $n = 1, 2, \ldots, m$, with fixed inputs of weight one. The output layer uses variable weights representing the Fourier coefficients a_n and b_n. No biases are included.

fully connected neural network includes connections between all nodes in consecutive layers, each with a distinct weight. In contrast, a convolutional network uses sparse connections and shared weights that allow it to represent transformations that are equivalent to the mathematical operation known as a convolution. In our network, we only include connections between nodes that correspond to the same phase difference $\Delta\theta_{i,j,k}$ and use weights that are independent of the indices i, j, and k. The weights in the first hidden layer are fixed at one, and the biases are fixed at zero. In the output layer, each harmonic is assigned a fixed bias of zero and a variable weight a_n or b_n corresponding to the Fourier coefficients. Figure 1 provides a schematic of this transformation as a neural network. Once the coupling terms have been computed, the rest of Eq. (4) and the resulting loss function [Eq. (6)] are straightforward to compute using vectorized operations.

We initialize the inferred parameters as follows: the adjacency matrix A has initial entries drawn from $\mathcal{N}(0.5, 1/N)$, the frequencies ω_k are initialized from $\mathcal{N}(0, 1/N)$, the coupling strength K is drawn from $\mathcal{N}(1, 1/N)$, and the Fourier coefficients $\boldsymbol{a}, \boldsymbol{b}$ of the coupling function are initialized at 0.

TensorFlow maintains a computational graph for these operations, which allows one to automatically compute gradients. One can, therefore, use gradient descent methods to compute the optimal estimates $(\hat{a}_n, \hat{b}_n, \hat{\omega}_k, \hat{A}_{kj}, \hat{K})$ for the parameters. We used a mini-batch gradient descent method with a batch size of 100; i.e., we randomly assign the data to batches of 100 time steps. For each batch, we calculate the gradient of the loss function and update the estimated parameters by taking a small step in a direction determined from the gradient. Once this has been repeated for all batches (one epoch), we pass through the data again for a total of 200 or more epochs. We determine the number of epochs experimentally by iterating until the mean squared error no longer decreases. Typically, 300 epochs are sufficient. However, for certain sweeps examining the role of $t_{\max}$, N_{res}, and σ, the number of epochs required for convergence is as high as 10 000.

By using small random batches of time steps, the gradient can be estimated quickly. This has the added benefit of introducing stochasticity into gradients, which makes the algorithm less susceptible to

getting caught in local minima. We use AdamOptimizer, which is a gradient descent method with momentum and an adaptive learning rate,[45] with default parameter values for optimization in TensorFlow.

There is minor variability in the success of the algorithm due to the random initialization of the inferred parameters and to the batching of the observed data. In order to ensure an accurate network reconstruction, we can retrain the model with new initial values several times and choose the result that has the smallest mean squared error for the velocity predictions on the validation set, which was not considered during the training process. For any individual model network, this validation error is correlated with the accuracy of the inferred parameters. In all examples below, we attempt to reconstruct each network five times before choosing the best reconstruction.

D. Postprocessing

The method described in Sec. II C produces continuous estimates for the parameters $(\hat{a}_n, \hat{b}_n, \hat{\omega}_k, \hat{A}_{kj}, \hat{K})$, which could be used to predict the dynamics of the system under a variety of initial conditions. However, we are interested in evaluating whether these parameter estimates are an accurate reconstruction of the original model. Before these values can be compared to the model parameters used to generate the data, additional postprocessing is needed. We, therefore, redefine our parameter estimates via the transformations (in order) $\hat{K}\hat{A}_{kj} \to \hat{A}_{kj}$, $c_0 + c_1\hat{\Gamma} \to \hat{\Gamma}$, where c_0 and c_1 are selected to minimize $\int_0^{2\pi} |\Gamma(\theta) - \hat{\Gamma}(\theta)|d\theta$, and $\hat{\omega}_k - \frac{c_0}{c_1 N}\sum_{j=1}^{N} \hat{A}_{kj} \to \hat{\omega}_k$. These transformations are necessary to address the aforementioned identifiability issues with K and to allow Γ to have a nonzero mean.

In the original model, the adjacency matrix entries are either zero or one. However, we treat the entries as continuous variables during optimization and then choose a threshold ϵ so that $A_{kj} < \epsilon$ is chosen to be 0 and $A_{kj} \geq \epsilon$ is chosen to be 1. In practice, one could fix ϵ or select ϵ so that the reconstructed model minimizes the mean squared error for the data set. However, we explore a range of threshold values using receiver operating characteristic (ROC) curves, which measure the quality of a classifier independent of any particular threshold (see Appendix B). We also report the reconstruction error rates using the value of ϵ that yields the largest F_1 score for the adjacency matrix reconstruction. The F_1 score is the harmonic mean of precision and recall and has a maximum value of 1, which corresponds to perfect reconstruction (see Appendix B). These optimal values are robust, and good performance is typically obtained over a wide range of thresholds (see Sec. IV for details).

III. EXPERIMENTAL DESIGN

In order to validate the robustness of our approach, we test the reconstruction method on data simulated with a variety of networks and parameter values. We explore this high-dimensional parameter space by fixing all of the model parameters except for one and then sweeping the remaining parameter over a wide range of values. See Tables I and II for a list of default values as well as the ranges considered. For each set of parameter values, we compute 30 networks with random initial phases and intrinsic frequencies. We then attempt to reconstruct the underlying model for each network and calculate various performance metrics for each reconstruction.

A. Coupling function

We evaluate the inferred coupling function $\hat{\Gamma}(\theta)$ by comparing it to the true coupling function $\Gamma(\theta)$ via the "normalized difference in area" defined as follows:

$$\text{Normalized difference in area} = \frac{\int_0^{2\pi} |\Gamma(\theta) - \hat{\Gamma}(\theta)|d\theta}{\int_0^{2\pi} |\Gamma(\theta)|d\theta}. \quad (7)$$

This represents the area between the true and estimated coupling function curves, weighted by the area under the curve of the true function. This quantity serves as a measure of the error in the reconstruction. A perfect reconstruction would have a normalized difference in area of zero. The initial estimate $\hat{\Gamma}(\theta) = 0$ corresponds to a value of 1. Therefore, values significantly lower than 1 indicate progress toward a correct reconstruction.

B. Intrinsic frequencies

We compare the inferred intrinsic frequencies $\hat{\omega}_k$ to the true intrinsic frequencies ω_k using the "mean absolute deviation" defined as follows:

$$\text{Mean absolute deviation} = \frac{1}{N}\sum_{k=1}^{N} |\omega_k - \hat{\omega}_k|. \quad (8)$$

Values near zero indicate accurate reconstructions. We considered alternative metrics such as relative deviations and the correlation between true and inferred frequencies, but these were less informative because they tend to amplify errors when the intrinsic frequencies are close to zero.

C. Adjacency matrix

We investigate several methods for evaluating the success of adjacency matrix reconstruction. The accuracy can be inspected visually by plotting the true and reconstructed matrices along with the absolute differences in a grid where values range from 0 (black) to 1 (white); see Figs. 2(a)–2(c). Since we are primarily interested in discrete adjacency values, we interpret reconstruction as a classification problem and compute three standard evaluation metrics: the F_1 score, the classification "error rate" for the connections, and the "area under the ROC curve." See Appendix B for precise definitions of these metrics.

IV. RESULTS

Our method can successfully reconstruct models with a variety of coupling functions. Figure 3(a) and Table V show the results for four different coupling functions (see Appendix A for details), three of which come from popular coupled oscillator models: the Kuramoto model, the Kuramoto-Sakaguchi model, and a phase reduction of the Hodgkin-Huxley model, and a fourth, a square wave, which represents a generic discontinuous function with high frequency Fourier harmonics. For the square-wave coupling function, the accuracy of the reconstruction suffers since the reconstruction includes only five Fourier harmonics, and the true coupling function contains an infinite number of harmonics with amplitudes
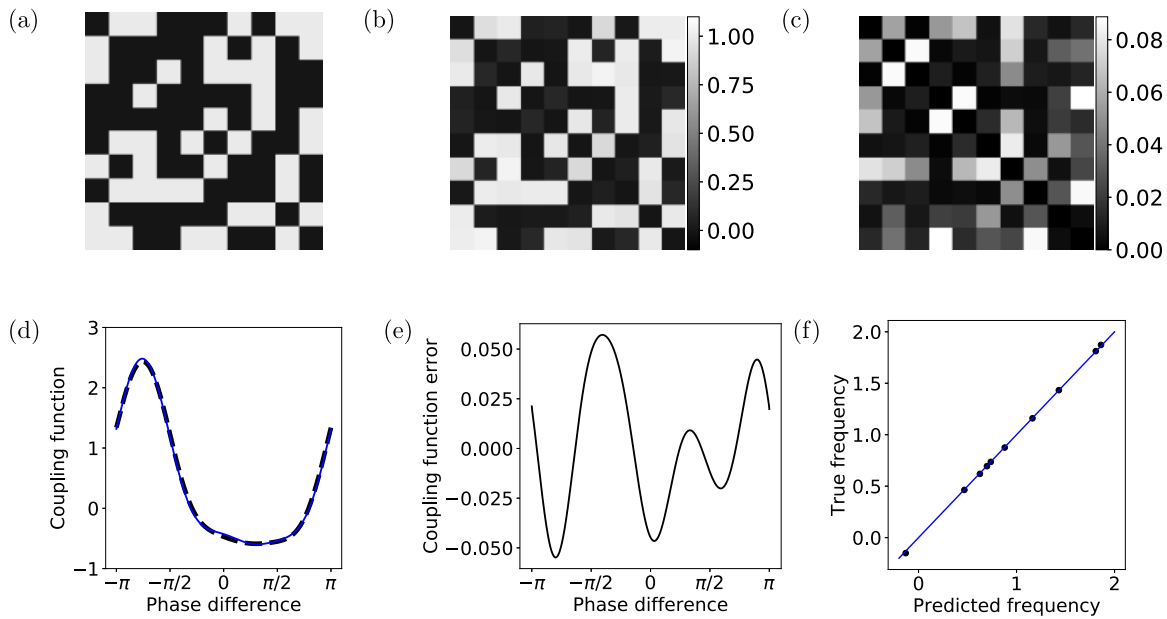
FIG. 2. Predicted adjacency matrix, coupling function, and frequencies. (a) True adjacency matrix, (b) predicted adjacency matrix, (c) absolute difference between true and predicted adjacency matrices, (d) coupling functions, true (blue, solid) and predicted (black, dashed), (e) coupling function difference, and (f) predicted vs true frequencies (black, dots) and $y = x$ (blue, solid). For this example, we obtain an adjacency matrix classification error rate of 0 for thresholds between 0.1 and 0.9, a normalized difference in area of 0.032 for the coupling function, and a mean absolute deviation of 0.008 for the estimated natural frequencies.

that decay slowly due to the discontinuities. Despite these limitations, the adjacency matrix was still estimated with a high degree of accuracy.

Counterintuitively, as the number of oscillators increases [Fig. 3(b) and Table VI], we find that reconstruction accuracy of the coupling function improves initially (from $N = 5$ to $N = 20$). The negative slope in the normalized difference in error is statistically significant with a p-value of 0.0001. This can be explained by the observation that although increasing the number of oscillators increases

the number of unknown parameters, it also provides a greater number of pairwise phase differences, which can aid in reconstruction of the coupling function.

In Fig. 4 and Table VII, we consider simulations where the standard deviation σ of the oscillator frequencies ranges from 0.01 to 1. The normalized difference in area of the coupling function and the mean absolute deviation of the inferred frequencies both confirm that we can infer the network successfully for $\sigma \leq 1$. For larger standard deviations, the quality of the reconstruction suffers slightly,
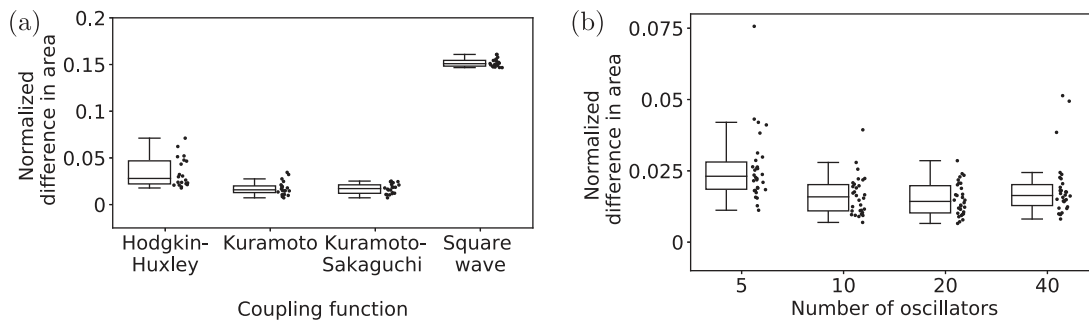
FIG. 3. Performance of coupling function reconstruction for different coupling functions and varying numbers of oscillators. (a) Normalized difference in area for different coupling functions. For the Hodgkin-Huxley, Kuramoto, and Kuramoto-Sakaguchi coupling functions, the normalized difference in area has the order $O(10^{-2})$. For a square-wave coupling function, the normalized difference in area is larger but still reasonably small [$O(10^{-1})$]. Whiskers are 1.5 times the interquartile range (IQR). (b) Normalized difference in area [Eq. (7)] for different number of oscillators N. As the number of oscillator increases, the normalized difference in area remains small. The result is shown for the Kuramoto model.

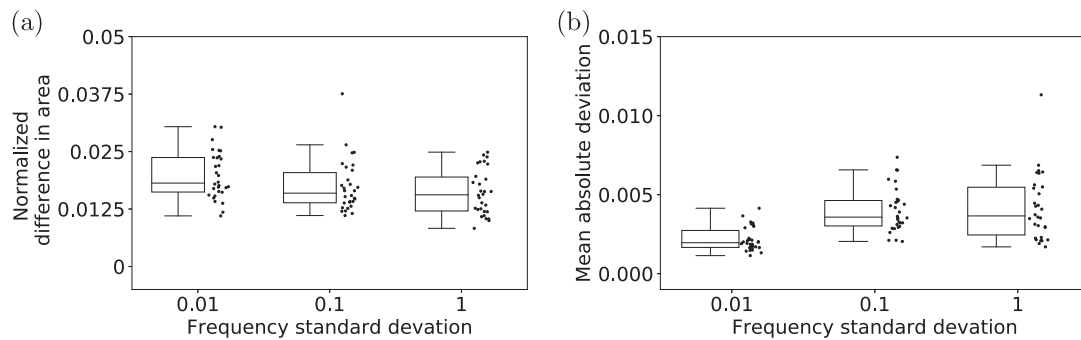**FIG. 4.** Performance metrics of the coupling function and the intrinsic frequencies for various frequency standard deviations. (a) Normalized difference in area. As the frequency standard deviation changes from 0.01 to 1, the normalized difference in area remains small. (b) Mean absolute deviation of the frequencies [Eq. (8)]. A small nonzero deviation in the intrinsic frequencies tends to reduce the mean absolute deviation. Here, the result is shown for the Kuramoto model.

since the large frequencies dominate the coupling terms within the phase velocity relationship.

We also explore the impact of noise on the model reconstruction since both observation noise and dynamic noise would be present in experiments. As in Ref. 18, we introduce dynamic noise using stochastic differential equations with Gaussian noise. We also add Gaussian noise to the observed phases after integrating the governing differential equations. In Tables VIII and IX, we demonstrate that our model reconstruction method is robust to moderate amounts of both types of noise. For both types of noise, we use mean 0 and standard deviations between 0 and 1. Figure 5 shows that dynamic noise with a standard deviation less than 10^{-3} helps with the reconstruction of both coupling function and frequencies. This can be explained by the fact that a small amount of noise keeps the system from reaching equilibrium, effectively increasing the duration of the transients that provide useful information about the structure of the network.

We carry out additional parameter sweeps with varying network connectivity p (Table X), maximum simulation time $t_{\max}$ (Table XI), and the number of simulation restarts (Table XII) with all other parameters set using the default values given in Tables I–III. In each

case, we are consistently able to reconstruct the coupling function, the underlying network, and the intrinsic frequencies over a wide range of model parameters. Tables V–XVII in Appendix C illustrate averaged numerical results for 30 trials of each of these parameter sweeps in terms of evaluation metrics such as normalized difference in area, mean absolute deviation, error rate, area under the ROC curve, and a range of thresholds that yield F_1 scores above 90% of the largest value.

A. Results for perturbations of synchronous dynamics

As Refs. 18 and 29 point out, when a network remains synchronized, i.e., when $\dot{\theta}_1 = \dot{\theta}_2 = \cdots = \dot{\theta}_N$, one cannot infer the model parameters due to the fact that the observed phases no longer provide linearly independent equations. Furthermore, even with precise knowledge of the adjacency matrix, one could not hope to reconstruct the coupling function $\Gamma(\theta_k - \theta_j)$ without data over a wide range of phase differences $\theta_k - \theta_j$.

We investigate a method for using small perturbations to introduce brief transients into the dynamics to allow for successful model reconstruction. To test this, we use nearly identical oscillators with
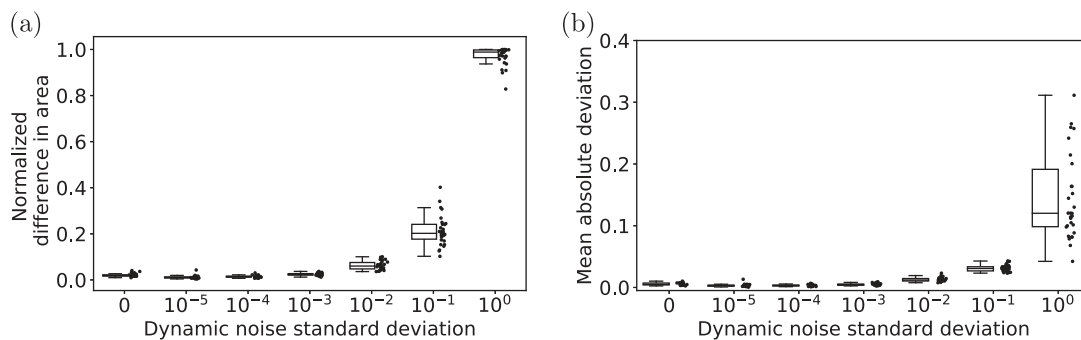
FIG. 5. Response of reconstruction to dynamic noise with various standard deviations. (a) Normalized difference in area of the coupling function. When the dynamic noise standard deviation is less than 10^{-3}, the normalized difference in area is smaller than for a model without noise. As the dynamic noise standard deviation increases to 10^{-1}, the normalized difference in area increases. A significant increase is observed as the noise standard deviation increases to 1. (b) Mean absolute deviation of the frequencies. The mean absolute deviation remains small as the dynamic noise standard deviation increases to 10^{-1}. Here, the result is shown for the Kuramoto model.

FIG. 6. Reconstruction of a model with synchronous dynamics using random perturbation or phase resetting. Left: the normalized difference in area for the coupling function. Middle: the error rate for the adjacency matrix. Right: the mean absolute deviation for the intrinsic frequencies. (a)–(c) Only one oscillator is perturbed repeatedly using random normally distributed phase perturbations $\eta_{pert} \sim \mathcal{N}(0, \sigma_{pert}^2)$ with $\sigma_{pert} = 0.01$. (d)–(f) A fixed subset of three oscillator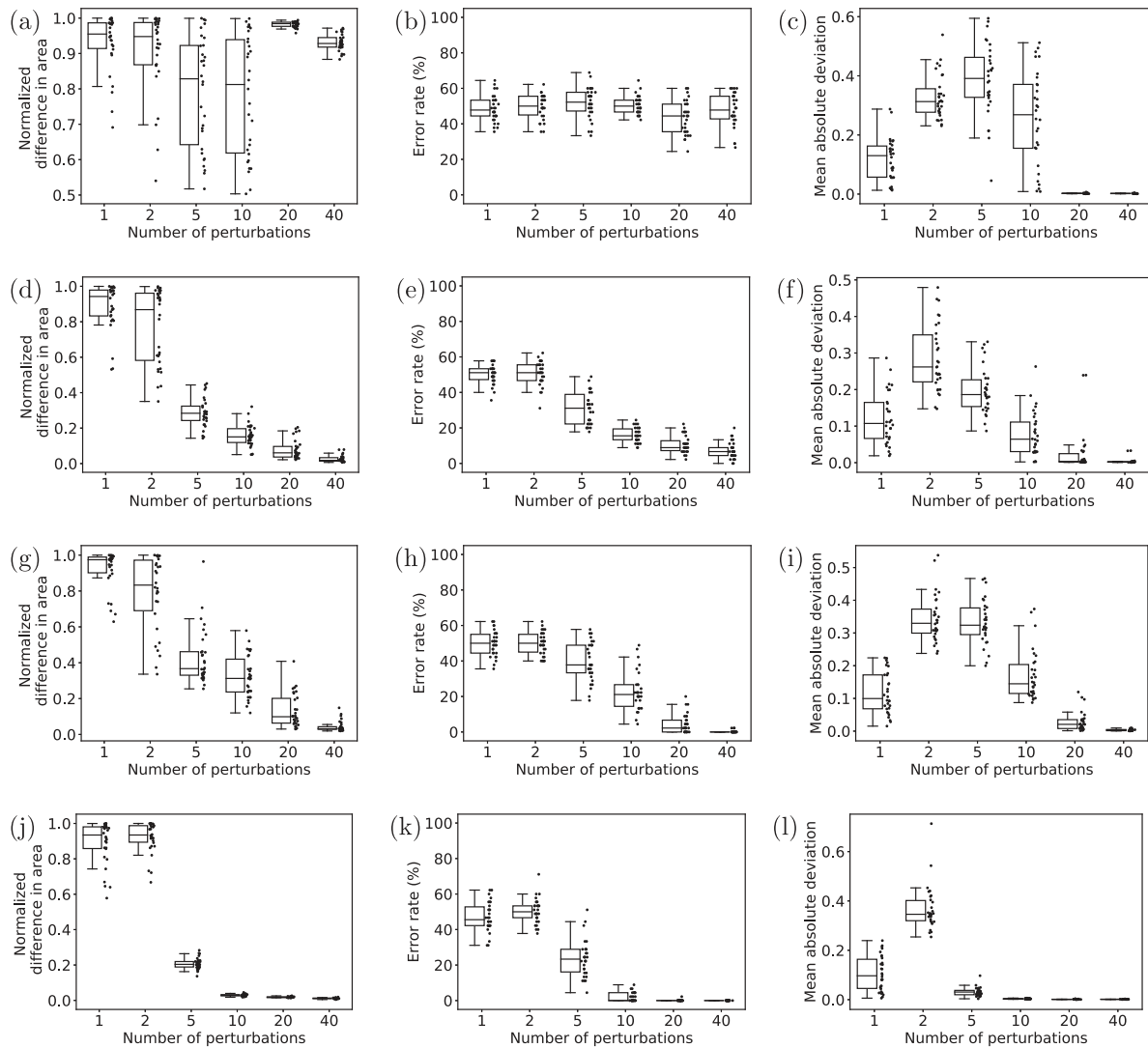s are perturbed repeatedly with a larger perturbation $\sigma_{pert} = 10$. (g)–(i) A single oscillator is selected randomly for each perturbation (with $\sigma_{pert} = 10$). (j)–(l) We reset a subset of three oscillators selected randomly with each perturbation. Here, the oscillators are nearly identical: $\sigma = 0.0001$ and with initial condition $\theta_k = 0$ for all k. We simulate the dynamics for $t_{max} = 200$ units of time and introduce perturbations at times $t = kt_{max}/N_{pert}$ for $k = 1, 2, \ldots, N_{pert}$. The result is shown for the Kuramoto model.

frequency standard deviation $\sigma = 0.0001$, which causes the system to quickly converge to a synchronized state for almost all initial conditions. We then initialize $\boldsymbol{\theta}(0) = \mathbf{0}$ so that the system begins from perfect synchrony. Then, at times kt_{max}/N_{pert} for $k = 1, 2, \ldots N_{pert}$, we add a phase perturbation to a subset of the oscillators. Each perturbation causes some of the oscillators to briefly become desynchronized. In this way, the total number of observed phases remains constant, but the fraction of those observations that occur during transient dynamics is proportional to the number of

perturbations, N_{pert}. The observed phases during these transients provide meaningful data about the structure of the network. Figure 6 demonstrates that, as expected, with too few or too small perturbations, model reconstruction is unsuccessful. However, as the number of perturbations N_{pert} increases, the accuracy of the estimated coupling function, adjacency matrix, and the oscillator frequencies improves.

We explore two methods for selecting which oscillators to perturb: fixed subsets, in which the subset of perturbed oscillators is
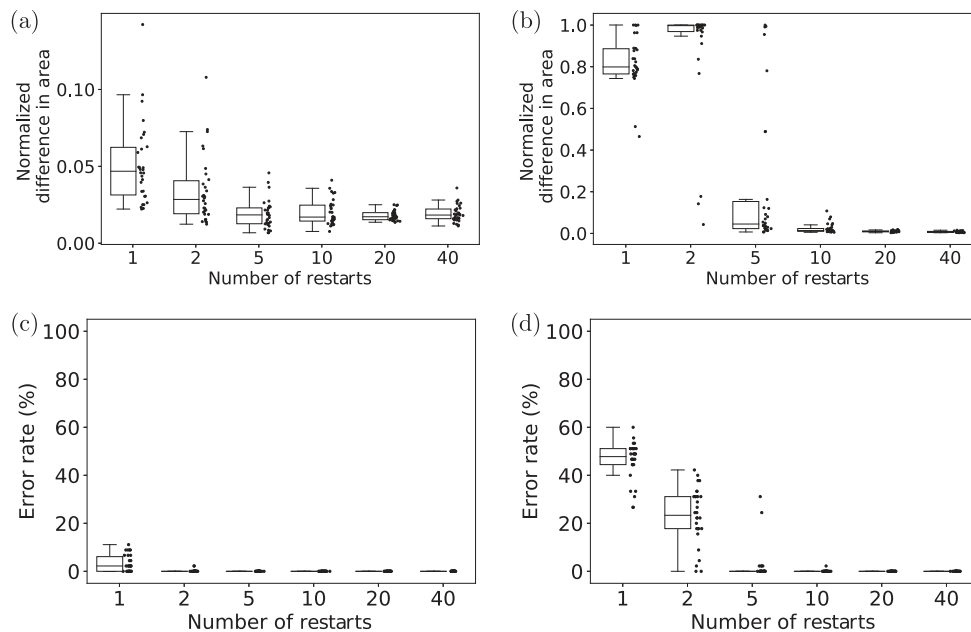
FIG. 7. Comparison of our method (left column) with the method described in Ref. 29 (right column) for different numbers of restarts. Panels (a) and (b) display the normalized difference in area of the coupling function. For the coupling function reconstruction, our method performs well with a small number of restarts, while Pikovsky's method requires a relatively larger number of restarts. Panels (c) and (d) display the error rate of the adjacency matrix. Again, both methods perform well with 10 or more restarts. Our method provides more accurate reconstructions when the number of restarts is five or less.

selected at the beginning of the experiment and these same oscillators are perturbed repeatedly, and random subsets, in which a random subset of the system is selected for each perturbation. Although a perturbation to the phase of a particular oscillator does propagate to the phases of neighboring oscillators through the coupling terms, those perturbations decay quickly and are, therefore, most effective for revealing the local structure of the network. As such, although perturbations to a fixed subset might be more practical in a physical experiment, one must typically perturb a larger fraction of the system to obtain comparable performance to that which is obtained with perturbations to random subsets.

We also consider two types of phase perturbations: phase resets, in which selected oscillators have their phases reset to 0, and phase shifts, in which selected oscillators have their phases modified by adding a random shift $\eta_{pert} \sim \mathcal{N}(0, \sigma_{pert}^2)$. Phase resets may be more feasible from an experimental perspective but have the drawback of preserving the mutual synchrony of the subset of oscillators that are perturbed. As such, one will typically need to use random subsets in tandem with phase resets in order to be able to resolve the connections between the perturbed oscillators.

In Figs. 6(a)–6(c), we use phase shifts with $\sigma_{pert} = 0.01$ to a fixed subset of size 1, i.e., a single oscillator out of 10. This gives poor reconstruction regardless of the number of perturbations due to the small size of the perturbation. On the other hand, in Figs. 6(d)–6(f), we perturb a fixed subset (3 out of 10 oscillators) with random phase shifts with $\sigma_{pert} = 10$. (These phase shifts are virtually indistinguishable from uniform perturbations $X \sim \mathcal{U}[-\pi, \pi]$.) In this case, the

perturbations affect a sufficiently large proportion of the oscillators, and performance begins to improve once there are 5 or more perturbations. The results in Figs. 6(g)–6(i) illustrate that one only needs to perturb 1 out of 10 oscillators to obtain similar performance when the oscillators selected for phase shifts are chosen randomly. In Figs. 6(j)–6(l), we show the results using phase resets to random subsets of 3 oscillators. Again, performance begins to improve dramatically once 5 or more perturbations are used. The tables summarizing the results of these perturbation strategies are provided in Appendix C.

B. Comparison with previous work

The amount of transient data necessary for reconstruction is also useful in comparing our method to previous approaches. As discussed in Sec. II B, Pikovsky proposes an alternative method where distinct coupling functions are considered for the interaction of each pair of oscillators.[29] This ensures that the system of equations is linear; however, it also increases the number of unknown coefficients significantly. Our approach relies on the assumption that the same coupling function Γ is used for all pairs of oscillators. This is a reasonable approximation for physical systems where the physical mechanisms governing the interactions between oscillators are the same.

Figure 7 shows that when the coupling functions are identical, our method provides more accurate reconstructions of both the network topology and the coupling function with smaller amounts

of transient data. As the amount of transient data increases, both methods ultimately achieve perfect network reconstruction, while the approach in Ref. 29 ultimately obtains slightly better estimates for the coupling function. Therefore, our approach would be preferred under circumstances where the amount of data is limited and when the coupling functions are nearly identical.

V. DISCUSSION AND CONCLUSIONS

In this paper, we have designed a method for reconstructing models of coupled oscillator networks including both the network connections and the intrinsic oscillator properties. We hope analysts might investigate (non)convexity of our penalty function and properties of minimizers. These issues aside, after testing our method with many different parameters, we conclude that our algorithm can successfully infer the model.

A common challenge that our method and many related ones encounter is that the procedure fails if the system synchronizes too quickly. One remedy is to have a sufficient number of observations during the desynchronized transients. As we demonstrate, one can ensure that these data are available by repeating experiments with multiple initial conditions or by adjusting the model parameters to inhibit synchronization by instituting large frequency variability (σ) or dynamic noise. An alternative remedy is to move the system away from synchrony using perturbations, preferably ones that are physically realizable. The introduction of these perturbations provides useful transient data. By perturbing a sufficiently large subset of oscillators with a large enough change, we are able to infer the model accurately. Our hope is that this method will be adopted by experimentalists and used with experimental data to aid with the construction of interpretable models for the dynamics of networks of coupled oscillators.

Although the numerical experiments outlined here involve Erdös-Rényi networks, the method can be applied more generally to a broad class of networks. Preliminary testing suggests that similar results can be obtained for other topologies such as star, small-world, scale-free networks, and clique networks.

It is also straightforward to extend this approach to other models for coupled phase oscillators such as the Winfree model[46] or even more general oscillator models such as the Stuart-Landau model[47] in which both phase and amplitude variations are permitted. Indeed, any system where the unknown functions are periodic can be represented using our technique.

For models containing unknown functions that are not periodic such as the Hodgkin-Huxley model,[48] a Fourier series representation is not possible. In these cases, one could represent the coupling functions using feed-forward neural networks, which are capable of representing continuous functions to arbitrary accuracy using a finite number of parameters.[49] Given a sufficiently rich data set, one could still use our approach with backpropagation to learn the structure of the neural network approximation for the coupling function.

APPENDIX A: COUPLING FUNCTIONS

The coupling functions investigated are described in Table IV. The first three classes of functions were selected due to their use in popular coupled oscillator models. The last was an example of a generic coupling function with higher order harmonics that was tested to verify that reconstruction of the adjacency matrix is possible even with an imperfect approximation to the coupling function.

APPENDIX B: EVALUATION METRICS FOR CLASSIFICATION

The F_1 score is defined as

$$\left(\frac{\text{precision}^{-1} + \text{recall}^{-1}}{2}\right)^{-1} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}.$$

Here, precision is the fraction of true positives among all inferred positives, while recall is the fraction of true positives among all positives. The F_1 score is a value between 0 and 1; when reporting the

TABLE IV. Coupling functions used in parameter sweeps described in Sec. III.

Name	Function	Reference
Kuramoto	$\sin(\theta)$	11
Kuramoto-Sakaguchi	$\sin(\theta - 0.1)$	35
Hodgkin-Huxley	$0.383 + 1.379\sin(\theta + 3.93) + 0.568\sin(2\theta + 0.11) + 0.154\sin(3\theta + 2.387)$	36
Square wave	$\dfrac{\sin(\theta - \pi/4)}{|\sin(\theta - \pi/4)|}$	N/A

TABLE V. Sweep through the coupling function Γ.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width ($> 90\%$)
Γ	Kuramoto	0.0175 ± 0.0075	0.004 ± 0.001	0.0 ± 0.0	1.0 ± 0.0	0.4338
Γ	Kuramoto-Sakaguchi	0.0169 ± 0.0054	0.005 ± 0.002	0.0 ± 0.0	1.0 ± 0.0	0.3924
Γ	Hodgkin-Huxley	0.0343 ± 0.0159	0.011 ± 0.004	0.0 ± 0.0	1.0 ± 0.0	0.7054
Γ	Square wave	0.1518 ± 0.0044	0.006 ± 0.002	0.0 ± 0.0	1.0 ± 0.0	0.4059

TABLE VI. Sweep of the number of oscillators N.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width ($> 90\%$)
N	5	0.026 ± 0.0125	0.0052 ± 0.003	0.0 ± 0.0	1.0 ± 0.0	0.1193
N	10	0.0164 ± 0.007	0.0047 ± 0.0031	0.0 ± 0.0	1.0 ± 0.0	0.4057
N	20	0.015 ± 0.0057	0.0046 ± 0.0016	0.140 ± 0.601	0.9997 ± 0.0016	0.3041
N	40	0.0189 ± 0.0104	0.0045 ± 0.0011	1.231 ± 1.018	0.9946 ± 0.0053	0.223

TABLE VII. Sweep of the standard deviation of the oscillator frequencies σ.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width ($> 90\%$)
σ	0.01	0.0198 ± 0.0052	0.0022 ± 0.0008	0.0 ± 0.0	1.0 ± 0.0	0.8142
σ	0.1	0.0175 ± 0.0057	0.004 ± 0.0014	0.0 ± 0.0	1.0 ± 0.0	0.8095
σ	1	0.0161 ± 0.0049	0.0042 ± 0.0021	0.0 ± 0.0	1.0 ± 0.0	0.7975

TABLE VIII. Sweep of the level of observation noise.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width ($> 90\%$)
Noise	0	0.0193 ± 0.0098	0.0048 ± 0.002	0.0 ± 0.0	1.0 ± 0.0	0.4204
Noise	1×10^{-05}	0.0202 ± 0.0108	0.0053 ± 0.0024	0.0 ± 0.0	1.0 ± 0.0	0.3845
Noise	0.0001	0.018 ± 0.0076	0.0048 ± 0.0023	0.0 ± 0.0	1.0 ± 0.0	0.4365
Noise	0.001	0.0186 ± 0.0081	0.0053 ± 0.002	0.0 ± 0.0	1.0 ± 0.0	0.3956
Noise	0.01	0.0186 ± 0.0079	0.0052 ± 0.0025	0.0 ± 0.0	1.0 ± 0.0	0.4135
Noise	0.1	0.0385 ± 0.0138	0.0086 ± 0.0021	0.0 ± 0.0	1.0 ± 0.0	0.3544
Noise	1	0.831 ± 0.1343	0.118 ± 0.0629	46.296 ± 8.579	0.5122 ± 0.0741	0.0472

TABLE IX. Sweep of the level of noise in system dynamics.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width ($> 90\%$)
dyn_noise	0	0.0211 ± 0.0067	0.0055 ± 0.0018	0.0 ± 0.0	1.0 ± 0.0	0.3814
dyn_noise	1×10^{-05}	0.0117 ± 0.0069	0.0032 ± 0.0023	0.0 ± 0.0	1.0 ± 0.0	0.4048
dyn_noise	0.0001	0.0146 ± 0.0049	0.0032 ± 0.0013	0.0 ± 0.0	1.0 ± 0.0	0.411
dyn_noise	0.001	0.0241 ± 0.0056	0.0049 ± 0.0017	0.0 ± 0.0	1.0 ± 0.0	0.4517
dyn_noise	0.01	0.0633 ± 0.0189	0.0128 ± 0.0034	0.370 ± 1.025	0.9982 ± 0.0072	0.3282
dyn_noise	0.1	0.2136 ± 0.065	0.0312 ± 0.0055	14.148 ± 5.003	0.8903 ± 0.0496	0.2224
dyn_noise	1	0.9716 ± 0.0399	0.1552 ± 0.0971	42.444 ± 8.953	0.5177 ± 0.1004	0.0406

TABLE X. Sweep of the network connectivity parameter p.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width (> 90%)
p	0.1	0.1199 ± 0.2314	0.0054 ± 0.0047	3.111 ± 14.97	0.9688 ± 0.1548	0.3387
p	0.2	0.023 ± 0.0098	0.004 ± 0.0022	0.074 ± 0.41	0.9999 ± 0.0006	0.5222
p	0.3	0.0186 ± 0.0073	0.0047 ± 0.0025	0.0 ± 0.0	1.0 ± 0.0	0.5175
p	0.4	0.0207 ± 0.0089	0.0047 ± 0.0022	0.0 ± 0.0	1.0 ± 0.0	0.5167
p	0.5	0.0189 ± 0.0097	0.0051 ± 0.0024	0.0 ± 0.0	1.0 ± 0.0	0.4669
p	0.6	0.0183 ± 0.0072	0.006 ± 0.0019	0.0 ± 0.0	1.0 ± 0.0	0.349
p	0.7	0.0169 ± 0.0071	0.005 ± 0.0025	0.0 ± 0.0	1.0 ± 0.0	0.3227
p	0.8	0.0194 ± 0.0077	0.0057 ± 0.0026	0.0 ± 0.0	1.0 ± 0.0	0.2725
p	0.9	0.0289 ± 0.0271	0.0077 ± 0.0056	0.0 ± 0.0	1.0 ± 0.0	0.2739

TABLE XI. Sweep of the simulation time t_{max} (duration of each transient).

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width (> 90%)
t_{max}	2	0.0204 ± 0.011	0.0042 ± 0.002	0.0 ± 0.0	1.0 ± 0.0	0.8225
t_{max}	5	0.0185 ± 0.0064	0.0047 ± 0.0022	0.0 ± 0.0	1.0 ± 0.0	0.8186
t_{max}	10	0.0193 ± 0.0066	0.004 ± 0.0013	0.0 ± 0.0	1.0 ± 0.0	0.8175
t_{max}	20	0.0192 ± 0.0053	0.0041 ± 0.0013	0.0 ± 0.0	1.0 ± 0.0	0.565
t_{max}	50	0.0191 ± 0.0093	0.0064 ± 0.0038	0.0 ± 0.0	1.0 ± 0.0	0.2964

TABLE XII. Sweep through the number of transients observed for a simulation that reinitializes all oscillator phases from $\mathcal{U}[0, 2\pi]$.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width (> 90%)
N_{res}	1	0.0513 ± 0.0266	0.0237 ± 0.0127	3.111 ± 3.531	0.9815 ± 0.027	0.447
N_{res}	2	0.0343 ± 0.0222	0.0082 ± 0.007	0.148 ± 0.564	0.9986 ± 0.0067	0.6943
N_{res}	5	0.019 ± 0.0094	0.0039 ± 0.002	0.0 ± 0.0	1.0 ± 0.0	0.806
N_{res}	10	0.0201 ± 0.0084	0.0042 ± 0.0028	0.0 ± 0.0	1.0 ± 0.0	0.808
N_{res}	20	0.018 ± 0.0033	0.0042 ± 0.0017	0.0 ± 0.0	1.0 ± 0.0	0.8295
N_{res}	40	0.0192 ± 0.0055	0.0041 ± 0.0016	0.0 ± 0.0	1.0 ± 0.0	0.8178

TABLE XIII. Sweep through the number of transients observed for a simulation that selects a fixed oscillator and adds a random Gaussian perturbation to its phase with standard deviation 0.01.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width (> 90%)
N_{pert}	1	0.9318 ± 0.082	0.118 ± 0.0719	48.667 ± 7.459	0.2934 ± 0.0999	0.0252
N_{pert}	2	0.9054 ± 0.1176	0.3284 ± 0.0747	49.778 ± 6.920	0.2904 ± 0.0783	0.0306
N_{pert}	5	0.7928 ± 0.1568	0.3853 ± 0.1243	51.852 ± 8.702	0.3281 ± 0.0895	0.0688
N_{pert}	10	0.7828 ± 0.1698	0.2566 ± 0.1579	50.593 ± 5.110	0.3843 ± 0.0697	0.1403
N_{pert}	20	0.9818 ± 0.0083	0.0024 ± 0.0011	43.630 ± 9.383	0.5153 ± 0.0959	0.1026
N_{pert}	40	0.931 ± 0.0234	0.0019 ± 0.0007	47.556 ± 9.403	0.4909 ± 0.0935	0.0931

TABLE XIV. Sweep through the number of transients observed for a simulation that selects a random oscillator and adds a random Gaussian perturbation to its phase with standard deviation 10.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width (> 90%)
N_{pert}	1	0.9193 ± 0.1111	0.1161 ± 0.0643	49.778 ± 7.042	0.3189 ± 0.1051	0.0298
N_{pert}	2	0.7919 ± 0.2112	0.343 ± 0.0724	49.852 ± 6.331	0.279 ± 0.0949	0.0409
N_{pert}	5	0.4298 ± 0.1535	0.3328 ± 0.0691	39.778 ± 11.032	0.5271 ± 0.1261	0.2041
N_{pert}	10	0.3211 ± 0.1187	0.1697 ± 0.0769	22.741 ± 10.975	0.7388 ± 0.0974	0.2111
N_{pert}	20	0.1311 ± 0.0931	0.0288 ± 0.0302	4.667 ± 5.296	0.9536 ± 0.0601	0.3584
N_{pert}	40	0.0447 ± 0.0311	0.0036 ± 0.0026	0.222 ± 0.679	0.9988 ± 0.0057	0.4824

TABLE XV. Sweep through the number of transients observed for a simulation that selects 3 fixed oscillators and adds a random Gaussian perturbation to their phases with standard deviation 10.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width (> 90%)
N_{pert}	1	0.8851 ± 0.1332	0.1203 ± 0.0716	50.0 ± 5.277	0.316 ± 0.1035	0.0293
N_{pert}	2	0.7678 ± 0.215	0.2857 ± 0.0916	50.963 ± 6.879	0.315 ± 0.0915	0.0326
N_{pert}	5	0.286 ± 0.0821	0.1973 ± 0.0625	30.963 ± 9.096	0.6136 ± 0.1118	0.1767
N_{pert}	10	0.1567 ± 0.0624	0.0788 ± 0.0609	15.704 ± 4.246	0.7659 ± 0.0863	0.3186
N_{pert}	20	0.0783 ± 0.056	0.0268 ± 0.06	10.815 ± 4.943	0.8699 ± 0.0777	0.4238
N_{pert}	40	0.0262 ± 0.0191	0.0041 ± 0.0078	7.185 ± 4.837	0.9391 ± 0.0565	0.5419

TABLE XVI. Sweep through the number of transients observed for a simulation that selects 3 random oscillators and resets their phases to 0.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width (> 90%)
N_{pert}	1	0.891 ± 0.1231	0.1058 ± 0.0706	46.889 ± 8.444	0.3337 ± 0.1156	0.0299
N_{pert}	2	0.9214 ± 0.088	0.3684 ± 0.0897	50.444 ± 7.031	0.2938 ± 0.0783	0.0474
N_{pert}	5	0.2075 ± 0.0319	0.0313 ± 0.0179	23.704 ± 10.379	0.7769 ± 0.0874	0.2803
N_{pert}	10	0.0294 ± 0.0061	0.0033 ± 0.0014	1.926 ± 2.592	0.9894 ± 0.0157	0.3855
N_{pert}	20	0.0184 ± 0.003	0.0015 ± 0.0007	0.074 ± 0.406	0.9999 ± 0.0008	0.522
N_{pert}	40	0.0116 ± 0.0028	0.0015 ± 0.0005	0.0 ± 0.0	1.0 ± 0.0	0.6206

TABLE XVII. Sweep through the number of transients observed for a simulation that carries out the optimization for a linear system as proposed in Ref. 29.

Param	Value	Normalized difference in area	Mean absolute deviation	Error rate	Area under the ROC curve	Interval width (> 90%)
N_{res}	1	0.8188 ± 0.1236	$8633.13 \pm 25\,993.8$	45.852 ± 8.159	0.5239 ± 0.0886	2012.5215
N_{res}	2	0.8895 ± 0.2663	3481.08 ± 9497.15	23.037 ± 11.635	0.7415 ± 0.1298	1.6593
N_{res}	5	0.2267 ± 0.3482	2.5746 ± 11.3136	2.148 ± 7.063	0.9756 ± 0.0733	0.75
N_{res}	10	0.0238 ± 0.0242	0.0061 ± 0.0078	0.074 ± 0.406	0.9979 ± 0.0114	0.8436
N_{res}	20	0.0099 ± 0.0037	0.0024 ± 0.001	0.0 ± 0.0	1.0 ± 0.0	0.9131
N_{res}	40	0.0077 ± 0.0033	0.0023 ± 0.001	0.0 ± 0.0	1.0 ± 0.0	0.9329

error rate for the reconstructed adjacency matrix, we use the threshold $\epsilon = \epsilon_{max}$ that corresponds to the largest F_1 score. We also determine the interval of thresholds that yield F_1 scores above 90% of this largest value. The width of this interval is reported in Tables V–XVII in Appendix C as "Interval width (>90%)."

The error rate is defined as the percentage of entries in A that are incorrectly classified. We report the error rate for the optimal threshold ϵ at the end of Sec. II D.

The ROC curve, or receiver operating characteristic, is a parametric curve that uses the classification threshold as a parameter and uses the true positive rate and the false positive rate as the variables.[50] The area under the ROC curve measures the quality of a classifier independent of any particular threshold. A value of $1/2$ is consistent with blind guesses, and a value of 1 indicates a perfect classifier since it implies the existence of a threshold for which the rate of true positives is 1 and the rate of false positives is 0. Our ROC curves were generated using the scikit-learn package in Python.[51]

APPENDIX C: TABLES OF RESULTS

Tables V–XVII report the averaged numerical results of the experimental parameter sweeps described in Sec. III and reported in Sec. IV. The performance metrics used are defined in Appendix B. Values given in the tables represent the mean plus or minus one standard deviation of the corresponding performance metric over the 30 trials run at that parameter value. We note that some of the distributions, such as the error rate (which is nonnegative by definition), are skewed.

REFERENCES

[1] W. Trautwein and D. G. Kassebaum, "On the mechanism of spontaneous impulse generation in the pacemaker of the heart," J. Gen. Phys. **45**, 317–330 (1961).

[2] J. I. Stagner, E. Samols, and G. C. Weir, "Sustained oscillations of insulin, glucagon, and somatostatin from the isolated canine pancreas during exposure to a constant glucose concentration," J. Clin. Invest. **65**, 939–942 (1980).

[3] G. Buzsaki and A. Draguhn, "Neuronal oscillations in cortical networks," Science **304**, 1926–1929 (2004).

[4] R. P. Bartsch, A. Y. Schumann, J. W. Kantelhardt, T. Penzel, and P. C. Ivanov, "Phase transitions in physiologic coupling," Proc. Natl. Acad. Sci. U.S.A. **109**, 10181–10186 (2012).

[5] J. Buck, "Synchronous rhythmic flashing of fireflies," Q. Rev. Biol. **63**, 265–289 (1988).

[6] M. R. Tinsley, S. Nkomo, and K. Showalter, "Chimera and phase-cluster states in populations of coupled chemical oscillators," Nat. Phys. **8**, 662–665 (2012).

[7] P. Hadley, M. Beasley, and K. Wiesenfeld, "Phase locking of Josephson-junction series arrays," Phys. Rev. B **38**, 8712–8719 (1988).

[8] A. E. Motter, S. A. Myers, M. Anghel, and T. Nishikawa, "Spontaneous synchrony in power-grid networks," Nat. Phys. **9**, 191–197 (2013).

[9] J. Pantaleone, "Synchronization of metronomes," Am. J. Phys. **70**, 992–1000 (2002).

[10] Z. Néda, E. Ravasz, T. Vicsek, Y. Brechet, and A.-L. Barabási, "Physics of the rhythmic applause," Phys. Rev. E **61**, 6987–6992 (2000).

[11] Y. Kuramoto, "Self-entrainment of a population of coupled non-linear oscillators," in *International Symposium on Mathematical Problems in Theoretical Physics* (Springer, 1975), pp. 420–422.

[12] J. A. Acebron, L. L. Bonilla, C. J. Perez Vicente, F. Ritort, and R. Spigler, "The Kuramoto model: A simple paradigm for synchronization phenomena," Rev. Mod. Phys. **77**, 137–185 (2005).

[13] A. Arenas, A. Diaz-Guilera, J. Kurths, Y. Moreno, and C. Zhou, "Synchronization in complex networks," Phys. Rep. **469**, 93–153 (2008).

[14] S. H. Strogatz, "From Kuramoto to Crawford: Exploring the onset of synchronization in populations of coupled oscillators," Physica D **143**, 1–20 (2000).

[15] M. J. Panaggio and D. M. Abrams, "Chimera states: Coexistence of coherence and incoherence in networks of coupled oscillators," Nonlinearity **28**, R67–R87 (2015).

[16] W.-X. Wang, Q. Chen, L. Huang, Y.-C. Lai, and M. A. F. Harrison, "Scaling of noisy fluctuations in complex networks and applications to network prediction," Phys. Rev. E **80**, 016116 (2009).

[17] H. X. Ta, C. N. Yoon, L. Holm, and S. K. Han, "Inferring the physical connectivity of complex networks from their functional dynamics," BMC Syst. Biol. **4**, 70 (2010).

[18] S. G. Shandilya and M. Timme, "Inferring network topology from complex dynamics," New J. Phys. **13**, 013004 (2011).

[19] E. S. C. Ching, P.-Y. Lai, and C. Y. Leung, "Extracting connectivity from dynamics of networks with uniform bidirectional coupling," Phys. Rev. E **88**, 042817 (2013).

[20] E. S. C. Ching, P.-Y. Lai, and C. Y. Leung, "Reconstructing weighted networks from dynamics," Phys. Rev. E **91**, 030801 (2015).

[21] W. Lin, Y. Wang, H. Ying, Y.-C. Lai, and X. Wang, "Consistency between functional and structural networks of coupled nonlinear oscillators," Phys. Rev. E **92**, 012912 (2015).

[22] M. Timme and J. Casadiego, "Revealing networks from dynamics: An introduction," J. Phys. A Math. Theor. **47**, 343001 (2014).

[23] G. Tirabassi, R. Sevilla-Escoboza, J. M. Buldú, and C. Masoller, "Inferring the connectivity of coupled oscillators from time-series statistical similarity analysis," Sci. Rep. **5**, 10829 (2015).

[24] E. Bianco-Martinez, N. Rubido, C. G. Antonopoulos, and M. Baptista, "Successful network inference from time-series data using mutual information rate," Chaos **26**, 043102 (2016).

[25] F. Alderisio, G. Fiore, and M. di Bernardo, "Reconstructing the structure of directed and weighted networks of nonlinear oscillators," Phys. Rev. E **95**, 042302 (2017).

[26] M. T. Angulo, J. A. Moreno, G. Lippner, A.-L. Barabási, and Y.-Y. Liu, "Fundamental limitations of network reconstruction from temporal data," J. R. Soc. Interface **14**, 20160966 (2017).

[27] B. J. Stolz, H. A. Harrington, and M. A. Porter, "Persistent homology of time-dependent functional networks constructed from coupled time series," Chaos **27**, 047410 (2017).

[28] R. Cestnik and M. Rosenblum, "Reconstructing networks of pulse-coupled oscillators from spike trains," Phys. Rev. E **96**, 012209 (2017).

[29] A. Pikovsky, "Reconstruction of a random phase dynamics network from observations," Phys. Lett. A **382**, 147–152 (2018).

[30] M. G. Leguia, Z. Levnajic, L. Todorovski, and B. Zenko, "Reconstructing dynamical networks via feature ranking," Chaos **29**, 093107 (2019).

[31] K. K. Liu, R. P. Bartsch, Q. D. Ma, and P. C. Ivanov, "Major component analysis of dynamic networks of physiologic organ interactions," in *Journal of Physics: Conference Series* (IOP Publishing, 2015), Vol. 640, p. 012013.

[32] R. P. Bartsch, K. K. Liu, A. Bashan, and P. C. Ivanov, "Network physiology: How organ systems dynamically interact," PLoS One **10**, e0142143 (2015).

[33] P. C. Ivanov, K. K. Liu, A. Lin, and R. P. Bartsch, "Network physiology: From neural plasticity to organ network interactions," in *Emergent Complexity from Nonlinearity, in Physics, Engineering and the Life Sciences* (Springer, 2017), pp. 145–165.

[34] H. Daido, "Order function and macroscopic mutual entrainment in uniformly coupled limit-cycle oscillators," Prog. Theor. Phys. **88**, 1213–1218 (1992).

[35] H. Sakaguchi and Y. Kuramoto, "A soluble active rotator model showing phase transitions via mutual entertainment," Prog. Theor. Phys. **76**, 576–581 (1986).

[36] D. Hansel, G. Mato, and C. Meunier, "Phase dynamics for weakly coupled Hodgkin-Huxley neurons," Europhys. Lett. **23**, 367 (1993).

[37] P. Erdös and A. Rényi, "On the evolution of random graphs," Publ. Math. Inst. Hung. Acad. Sci. **5**, 17–60 (1960).

[38] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, see tensorflow.org for "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems" (2015).

[39] F. A. Rodrigues, T. K. D. Peron, P. Ji, and J. Kurths, "The Kuramoto model in complex networks," Phys. Rep. **610**, 1–98 (2016).

[40] J. Dormand and P. Prince, "A family of embedded Runge-Kutta formulae," J. Comput. Appl. Math. **6**, 19–26 (1980).

[41] R. L. Honeycutt, "Stochastic Runge-Kutta algorithms. I. White noise," Phys. Rev. A **45**, 600–603 (1992).

[42] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures," Anal. Chem. **36**, 1627–1639 (1964).

[43] K. P. Murphy, *Machine Learning: A Probabilistic Perspective* (MIT Press, 2012).

[44] M. J. Panaggio, M.-V. Ciocanel, and L. Lazarus, see https://github.com/mpanaggio/coupled_oscillator_network_model_reconstruction/tree/published for "Coupled Oscillator Network Model Reconstruction" (2019).

[45] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization" (2014); e-print arXiv:1412.6980.

[46] A. T. Winfree, *The Geometry of Biological Time* (Springer Science & Business Media, 2001), Vol. 12.

[47] N. Nakagawa and Y. Kuramoto, "Collective chaos in a population of globally coupled oscillators," Prog. Theor. Phys. **89**, 313–323 (1993).

[48] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," J. Physiol. **117**, 500–544 (1952).

[49] K. Hornik, "Approximation capabilities of multilayer feedforward networks," Neural Netw. **4**, 251–257 (1991).

[50] T. Fawcett, "An introduction to ROC analysis," Pattern Recognit. Lett. **27**, 861–874 (2006).

[51] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res. **12**, 2825–2830 (2011).