Inferring synchronizability of networked heterogeneous oscillators with machine learning

Liang Wang,[1] Huawei Fan [ORCID],[2] Yafeng Wang,[3] Jian Gao,[4] Yueheng Lan,[4] Jinghua Xiao,[4] and Xingang Wang [ORCID][1,*]

[1]*School of Physics and Information Technology, Shaanxi Normal University, Xi'an 710062, China*
[2]*School of Science, Xi'an University of Posts and Telecommunications, Xi'an 710121, China*
[3]*Nonlinear Research Institute, Baoji University of Arts and Sciences, Baoji 721016, China*
[4]*School of Science, Beijing University of Posts and Telecommunications, Beijing 100876, China*

In the study of network synchronization, an outstanding question of both theoretical and practical significance is how to allocate a given set of heterogeneous oscillators on a complex network in order to improve the synchronization performance. Whereas methods have been proposed to address this question in the literature, the methods are all based on accurate models describing the system dynamics, which, however, are normally unavailable in realistic situations. Here, we show that this question can be addressed by the model-free technique of a feed-forward neural network (FNN) in machine learning. Specifically, we measure the synchronization performance of a number of allocation schemes and use the measured data to train a machine. It is found that the trained machine is able to not only infer the synchronization performance of *any* new allocation scheme, but also find from a huge amount of candidates the optimal allocation scheme for synchronization.

I. BACKGROUND

Many real-world systems can be represented by complex networks of coupled oscillators [1], in which an interesting phenomenon is that in some circumstances the oscillators can be self-organized into coherent states of synchronized motions [2]. For its important implications in the functionality and performance of many realistic systems, network synchronization has been broadly interested and extensively studied by researchers from different fields in the past decades [3–5]. In exploring network synchronization, one of the central questions concerns the improvement of network synchronization performance, namely synchronizability, by a slight change of the system properties [6], e.g., introducing shortcut links or adjusting the coupling schemes [7–14]. For complex networks consisting of heterogeneous oscillators, an additional approach to improve the network synchronizability is reallocating the oscillators according to the network topology [14–21]. Whereas strategies have been proposed in the literature on how to reallocate the oscillators optimally and the efficiency of the strategies have been justified in different systems, the existing studies rely on prior knowledge of the network models [15–21], including the network structure, the coupling function, and the oscillator dynamics, which is normally unattainable in practice. Moreover, while evidence suggests that there might exist a one-to-one correspondence between oscillator allocation and network synchronizability, the mapping function is too complicated to be given explicitly, and the search for the optimal allocation still relies on large-scale simulations. The purpose of our present paper is to introduce a model-free technique in machine learning, namely the feed-forward neural network (FNN) [22–24], to learn from measured data the mapping function between oscillator allocation and network synchronizability, and utilize the trained machine to find the optimal allocation for synchronization.

II. PROBLEM DESCRIPTION

We study the synchronization behaviors of coupled non-identical phase oscillators described by the generalized Kuramoto model [2]

$$\dot{\theta}_i = \omega_i + \frac{\varepsilon}{d_i} \sum_{j=1}^{N} a_{ij} \sin(\theta_j - \theta_i), \tag{1}$$

with $i, j = 1, ..., N$ the oscillator (node) indices and N the network size. θ_i and ω_i represent, respectively, the phase and natural frequency of the ith oscillator. The coupling relationship of the oscillators, i.e., the network structure, is encoded in the adjacency matrix $\mathbf{A} = \{a_{ij}\}_{N \times N}$, with $a_{ij} = a_{ji} = 1$ if there is a link between nodes i and j, and $a_{ij} = a_{ji} = 0$ otherwise. $d_i = \sum_j a_{ij}$ denotes the degree of node i, and ε is the uniform coupling strength. The network structure is generated by the Erdös-Rényi model with connecting probability p [1]. In our studies, we set $N = 20$ and $p = 0.5$. The natural frequencies of the oscillators are randomly chosen within the range $(-1, 1)$, which, once chosen, will be fixed, but the oscillators are allowed to be reallocated freely on the network. The total number of allocation schemes is huge ($N!$), and different allocations give different synchronization performance [14–21]. The central question we ask is, given that the synchronization performance of a number of allocation schemes is known, can we predict the synchronization performance of a new allocation scheme without knowing the network dynamics?
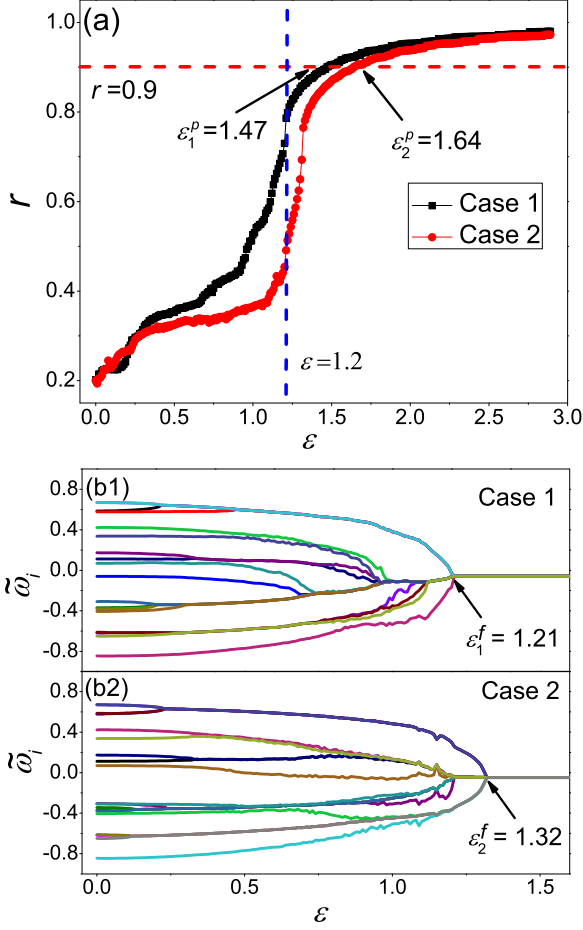
*wangxg@snnu.edu.cn

FIG. 1. Dependence of network synchronization performance on oscillator allocation. Shown are the transitions from desynchronization to phase (a) and frequency (b) synchronization for two random oscillator allocations on a random network containing $N = 20$ nodes. r: the synchronization order parameter; ε^p: the critical coupling where $r > r_c = 0.9$; ε^f: the critical coupling where the effective frequencies of the oscillators $\{\tilde{\omega}_i\}$ become identical.

To show the influences of oscillator allocation on synchronization performance, we choose two random allocations and, by solving Eq. (1) numerically, plot in Fig. 1(a) the variation of the synchronization order parameter r with respect to the coupling strength ε. Here, the synchronization order parameter is defined as $r = |\sum_j e^{i\theta_j}|/N$, with $j = 1, \ldots, N$ the oscillator index. We have $r \in (0, 1)$, with larger r representing stronger synchronization [1]. Figure 1(a) shows that the two order parameters are close for weak couplings ($\varepsilon < 0.5$), but are separated from each other for strong couplings [$\varepsilon \in (0.5, 2.0)$]. Defining phase synchronization as the point where r exceeds $r_c = 0.9$, the critical couplings for phase synchronization are $\varepsilon_1^p \approx 1.47$ and $\varepsilon_2^p \approx 1.64$ for the first and second allocations, respectively. Clearly, in terms of phase synchronization, the first allocation outperforms the second one. Denoting $\boldsymbol{\Omega}_l$ as a specific allocation of the oscillators and ε_l^p the corresponding critical coupling for phase synchronization, we have for each allocation a synchronization pair (data point) $(\boldsymbol{\Omega}_l, \varepsilon_l^p)$. Assuming that a number ($\tilde{M} \gg 1$) of such synchronization pairs are available (obtained by model

simulations or experiments), our *first objective* is to infer by machine the critical coupling for a new allocation to reach phase synchronization.

By the same allocation schemes studied in Fig. 1(a), we plot in Fig. 1(b) the transitions of the network dynamics from the perspective of frequency synchronization [1]. Frequency synchronization is featured by the identical effective frequencies of the oscillators. The effective frequency of oscillator i is $\tilde{\omega}_i = \langle \dot{\theta}_i \rangle_T$, with $\dot{\theta}_i$ the instant angular frequency of oscillator i and $\langle \cdots \rangle_T$ the time-average function. Figure 1(b) shows that for the first allocation, the effective frequencies of the oscillators become identical at the critical coupling $\varepsilon_1^f \approx 1.21$, while for the second allocation the critical coupling is $\varepsilon_2^f \approx 1.32$. Comparing to the results of phase synchronization shown in Fig. 1(a), we have $\varepsilon^f < \varepsilon^p$ for both allocations. This phenomenon is understandable, as the motions of the oscillators are more strongly correlated in phase synchronization than in frequency synchronization [4]. Assuming that a large number of frequency-based synchronization pairs of the form $(\boldsymbol{\Omega}_l, \varepsilon_l^f)$ are available, the *second objective* of our present work is to infer by machine the critical coupling ε^f for a new allocation scheme to achieve frequency synchronization.

Once the capability of the machine in inferring the synchronization performance of new allocation schemes has been confirmed, we shall then utilize the machine to find from a large number of candidates the optimal allocation giving the best synchronization performance, which is the *third objective* of our present work. In addition to the critical couplings (ε^p and ε^f), another approach to evaluate the synchronization performance of different allocations is comparing their synchronization order parameters under the same coupling strength. Taking the results in Fig. 1(a) as an example, at the intermediate-coupling strength $\varepsilon = 1.2$, the order parameter of the first allocation is about 0.8, while for the second allocation the order parameter is only about 0.5. Denoting $(\boldsymbol{\Omega}_l, r_l)_\varepsilon$ as the measured data in this case and assuming that a number of such measurements are available, the *fourth objective* of our present work is to infer by machine the order parameter of a new allocation scheme under a specific coupling strength.

III. FNN TECHNIQUE

The FNN adopted in our studies consists of three modules [22–24]: an input layer, a series of L hidden layers, and an output layer. The input layer is characterized by the matrix $\boldsymbol{W}_{\text{in}} \in \mathbb{R}^{n_1 \times N}$, which couples the input vector $\boldsymbol{\Omega}_l = [\omega_1, \ldots, \omega_N]^T$ (i.e., a specific allocation of the oscillators on the network) to the first hidden layer by the operation

$$\boldsymbol{x}_1 = f(\boldsymbol{W}_{\text{in}} \boldsymbol{\Omega}_l + \boldsymbol{b}_{\text{in}}). \tag{2}$$

Here, $\boldsymbol{x}_1 \in \mathbb{R}^{n_1}$ is the state vector of the first hidden layer, n_1 denotes the size of the first hidden layer, $\boldsymbol{b}_{\text{in}} \in \mathbb{R}^{n_1}$ is the bias vector associated to the input layer, and $f(x)$ is the activation function. The signals are then propagated to the other hidden layers in sequence as

$$\boldsymbol{x}_h = f(\boldsymbol{W}_h \boldsymbol{x}_{h-1} + \boldsymbol{b}_h), \tag{3}$$

where $h = 2, \ldots, L$ is the index of the hidden layers, $\boldsymbol{W}_h \in \mathbb{R}^{n_h \times n_{h-1}}$ is the coupling matrix between layers $h - 1$ and h,

n_h is the size of the hth hidden layer, and $\boldsymbol{b}_h \in \mathbb{R}^{n_h}$ is the bias vector associated to the hth hidden layer. After the Lth hidden layer is activated, we generate the output

$$y = f(\boldsymbol{W}_{\text{out}}\boldsymbol{x}_L + b_{\text{out}}), \tag{4}$$

with y a scalar, $\boldsymbol{W}_{\text{out}} \in \mathbb{R}^{1 \times n_L}$ the output matrix, and b_{out} the output bias. The hyperparameters of FNN are the number of hidden layers L and the number of nodes contained in each hidden layer $\{n_h\}$ with $h = 1, \ldots, L$. The model parameters of FNN include the coupling matrices ($\boldsymbol{W}_{\text{in}}$, $\boldsymbol{W}_h$, and $\boldsymbol{W}_{\text{out}}$) and the bias vectors ($\boldsymbol{b}_{\text{in}}$, $\boldsymbol{b}_h$, and b_{out}), which are to be learned from the measured data through a training process. In our studies, we adopt the all-to-all coupling strategy for nodes in the neighboring layers, and use the rectified linear activation function (ReLU) to update the node states [23].

The implementation of FNN consists of three phases: the training phase, the validation phase, and the prediction phase. In the training phase, the input data are the oscillator allocations $\{\boldsymbol{\Omega}_l\}_{l=1}^{M}$, and the mission of the training is to obtain the set of model parameters (i.e., the coupling matrices and bias vectors) such that the outputs given by Eq. (4) are as close as possible to the true values, which are the critical couplings (ε^p or ε^f) for objectives 1 and 2 and is the synchronization order parameter (r) for objective 4. In our studies, this is done by the conventional back propagation algorithm with the help of the ADAM (adaptive moment estimation) optimizer in TEN-SORFLOW [25]. After training, the performance of the machine is validated by a new data set different from the one used in the training phase. This process of training and evaluation is repeated for a number of rounds, while in each round the structure of FNN is reconstructed according to a unique set of hyperparameters. By the strategy of Bayesian optimization [23], we then find the set of hyperparameters giving the best performance on the validating data set, which completes the validation phase. Finally, in the prediction phase, with the optimal hyperparameters obtained in the validation phase and the model parameters obtained in the training phase, we drive the machine with a completely new oscillator allocation, and the output of the machine gives the prediction. (See Supplemental Material [26] for more details about FNN.)

IV. RESULTS

We start by showing the performance of FNN in inferring the critical coupling ε^p for phase synchronization. For simplicity, we consider here the scenario of constrained frequency allocations [19]. That is, only a fraction of the oscillators are reallocated on the network, while the locations of the other oscillators are fixed. The m relocatable oscillators are chosen by random, and the exchange of any pair of them generates a new allocation. In our studies, we set $m = 10$ and generate in total $\tilde{M} = 800$ random allocations. For each allocation, we first obtain by simulations the variation of the order parameter r with respect to the coupling strength ε by the increment $\Delta\varepsilon = 0.02$, and then find the critical coupling ε^p where r exceeds $r_c = 0.9$. We obtain in total $\tilde{M} = 800$ data points of the form $(\boldsymbol{\Omega}_l, \varepsilon_l^p)$, among which $M = 500$ points are used in the training phase for obtaining the model parameters and $M' = 300$ points are used in the validation phase for optimizing the machine hyperparameters (see Supplemental Material [26] for
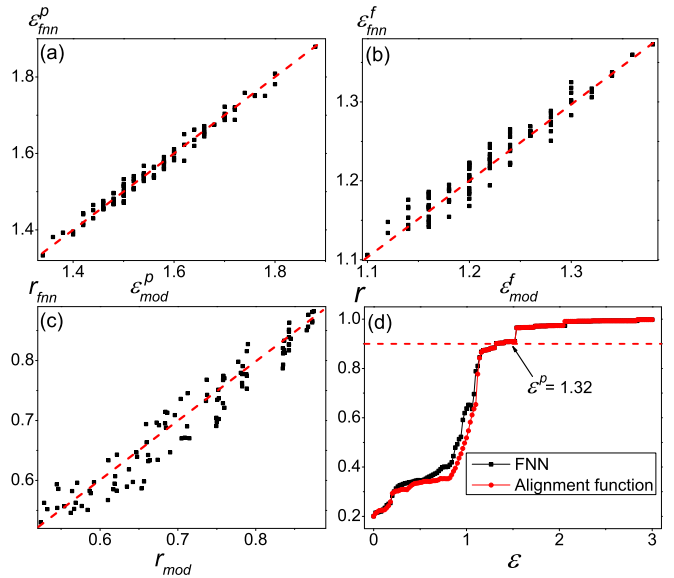

FIG. 2. Inferring network synchronizability with machine. (a) Inference of the critical coupling ε^p for phase synchronization. (b) Inference of the critical coupling ε^f for frequency synchronization. (c) Inference of the synchronization order parameter for the coupling strength $\varepsilon = 1.2$. ε_{fnn} and ε_{mod} denote, respectively, the results predicted by the machine and the results obtained by model simulations. Red dashed lines are diagonal lines. (d) Synchronization performance of the optimal allocations obtained by FNN (black squares) and by the method of alignment function (red disks).

the statistical properties of the data set and the influence of the data properties on machine performance). In this application, the optimal hyperparameters are $L = 1$ and $n_1 = 100$. That is, the FNN contains only one hidden layer and the size of the hidden layer is 100. Shown in Fig. 2(a) are the critical couplings predicted by the machine for 100 new allocations that are not included in the training and validating data set. We see that the results predicted by the machine are in good agreement with the results of model simulations. To evaluate the overall predicting performance of the machine, we introduce the normalized error $e^p = |1 - \varepsilon_{\text{fnn}}^p/\varepsilon_{\text{mod}}^p|$ for each prediction, with $\varepsilon_{\text{fnn}}^p$ and $\varepsilon_{\text{mod}}^p$ denoting, respectively, the critical couplings predicted by the machine and obtained by model simulation, and calculate the averaged predicting error $\langle e^p \rangle$ for $K = 1000$ new random allocations. We have $\langle e^p \rangle \approx 7 \times 10^{-3}$. Indeed, the machine is able to infer the critical coupling for phase synchronization with a high precision.

We next employ the FNN technique to predict the critical coupling for frequency synchronization. In doing this, we first regenerate the data set by model simulations, and then use the measured data to construct and train a new machine. The data point has the form $(\boldsymbol{\Omega}_l, \varepsilon_l^f)$, with $\boldsymbol{\Omega}_l$ the lth allocation scheme and ε_l^f the corresponding critical coupling. Again, we generate in total $\tilde{M} = 800$ data points ($M = 500$ points for training the model parameters and $M' = 300$ points for finding the optimal hyperparameters). In this application, the optimal hyperparameters are $L = 1$ and $n_1 = 100$. We consider still the scenario of constrained allocations, with the number of relocatable oscillators being $m = 10$. Figure 2(b) shows the

TABLE I. The performance of FNN in other network models. N: the network size. m: the number of relocatable oscillators. $\tilde{M}$: the number of data points obtained by model simulations in inferring phase synchronization. M: the number of data points used in machine training. $\langle e^p \rangle$: averaged prediction error for phase synchronization. $\langle e^f \rangle$: averaged prediction error for frequency synchronization. $\langle e \rangle_r$: averaged prediction error for synchronization order parameter under the coupling strength ε_r. Errors are averaged over $K = 1000$ random allocations.

Models	Nodal dynamics	N/m	$\tilde{M}/M$	$\langle e^p \rangle$	$\langle e^f \rangle$	$(\langle e \rangle_r, \varepsilon_r)$
Scale-free network [27]	Phase oscillator	100/10	1000/700	3.3×10^{-3}	4×10^{-3}	$(6 \times 10^{-2}, 1.38)$
Nepal power grid [28]	Phase oscillator	15/15	1500/1000	2.2×10^{-2}	3×10^{-2}	$(5 \times 10^{-2}, 1.0)$
Cat brain network [29]	Phase oscillator	53/10	1200/800	5×10^{-3}	6×10^{-3}	$(1.6 \times 10^{-2}, 1.3)$
Nematode neuronal network [30]	Phase oscillator	297/10	2000/1500	3.3×10^{-3}	2.2×10^{-3}	$(1.7 \times 10^{-2}, 1.48)$
Complex random network	Rössler oscillator [31]	20/10	2000/1500	1.1×10^{-2}	2.2×10^{-2}	$(2.7 \times 10^{-2}, 0.025)$
Complex ecological network [32]	Foodweb oscillator [33]	82/10	2500/2000	2.2×10^{-2}	2.2×10^{-2}	$(1.2 \times 10^{-2}, 0.5)$

prediction performance of the new machine for 100 random allocations. We see that the critical couplings are accurately predicted by the machine in all cases. In this application, the predicting error averaged over $K = 1000$ random allocations is about $\langle e^f \rangle = 1 \times 10^{-2}$, justifying hence the capability of the machine in inferring frequency synchronization.

Having shown the capability of the FNN technique in predicting phase and frequency synchronization, we continue to employ the technique as an efficient tool to find the optimal allocation for synchronization among a huge amount of candidates (objective 3 in our present work). To study, we first generate 10 000 allocations by exchanging the ($m = 10$) relocatable oscillators randomly, and then use the machine trained for predicting phase synchronization to estimate the critical coupling ε^p for each allocation. The optimal allocation is identified as the one with the smallest ε^p, which is about 1.32 among the 10 000 candidates. As the machine has been trained already and predictions are made by a mapping function, the search for the optimal allocation is accomplished in seconds. By numerical simulations, we plot in Fig. 2(d) the variation of r with respect to ε for the optimal allocation identified by the machine. It is seen that r exceeds $r_c = 0.9$ at $\varepsilon^p \approx 1.32$, which is exactly the critical coupling predicted by the machine. As a benchmark to evaluate the synchronization performance of the optimal allocation, we compare it to the optimal allocation obtained by the method of synchrony alignment function proposed in Ref. [19]. The alignment function is calculated as $J(\mathbf{\Omega}, \mathbf{L}) = (1/N) \sum_{i=2}^{N} \lambda_i^{-2} \langle \mathbf{v}_i, \mathbf{\Omega} \rangle^2$, with $\mathbf{L} = \mathbf{D} - \mathbf{A}$ the Laplacian matrix, $\mathbf{D}$ the diagonal matrix encoding the node degrees, $\{\lambda_i\}_{i=2}^{N}$ the nonzero eigenvalues of $\mathbf{L}$, and $\{\mathbf{v}_i\}_{i=2}^{N}$ the corresponding eigenvectors. We apply the function to the same set of candidates, and the one with the smallest value of J is identified as the optimal allocation. The synchronization performance of the new optimal allocation is also shown in Fig. 2(d). We see that the order parameters of the two optimal allocations are close for weak couplings ($\varepsilon < 1.2$) and are identical for strong couplings ($\varepsilon > 1.2$). An additional analysis has been conducted to check the properties of the optimal allocation identified by the machine, which shows that, similar to the results of the alignment function, the degrees and frequencies are positively correlated and the frequencies of the neighboring nodes are negatively correlated (see Supplemental Material [26] for details).

We finally employ the FNN technique to infer the synchronization order parameter (objective 4). Fixing $\varepsilon = 1.2$,

we obtain by simulations the order parameter for $\tilde{M} = 1000$ random allocations and, using $(\mathbf{\Omega}_l, r_l)_{l=1}^{\tilde{M}}$ as the inputs, optimize ($M' = 300$) and train ($M = 700$) a new machine. The hyperparameters of this new machine are $L = 3$ and $(n_1, n_2, n_3) = (150, 200, 50)$. The performance of the new machine in predicting the synchronization order parameter of 100 new random allocations is shown in Fig. 2(c). We see that the order parameters predicted by the machine agree with the true results very well. The normalized prediction error averaged over $K = 1000$ random allocations is $\langle e \rangle_r \approx 3.7 \times 10^{-2}$, which is slightly larger than the prediction errors of the critical couplings (7×10^{-3} for ε^p and 1×10^{-2} for ε^f). This arouses our interest about the dependence of prediction performance on network synchronization degree. Intuitively, the larger the order parameter, the stronger is the correlation between the oscillators, the easier it is for the machine to learn from the data the mapping function between oscillator allocation and network synchronizability, and the more accurate is the inference. To verify this conjecture, we analyze again the simulating results obtained in inferring phase synchronization (i.e., the synchronization transitions of $\tilde{M} = 800$ random allocations) and find the critical couplings where r exceeds $r_c = 0.6$. With the newly obtained data set, we construct and train a new machine (the hyperparameters are $L = 2$ and $n_1 = n_2 = 100$), and then use the trained machine to predict the order parameters for $K = 1000$ new random allocations. The averaged prediction error is $\langle e \rangle_r \approx 3 \times 10^{-2}$. To verify the conjecture further, we set $r_c = 0.3$ and check again the prediction performance (the hyperparameters are $L = 1$ and $n_1 = 100$). In this case, we have $\langle e \rangle_r \approx 0.2$. Indeed, the prediction performance is improved by increasing the synchronization degree.

V. DISCUSSIONS

The capability and performance of the FNN technique have been verified in other network models, including networks of different sizes and topological features, networks consisting of heterogeneous chaotic oscillators, and also the scenario of unconstrained allocations. The results are summarized in Table I (details are given in Supplemental Material [26]). We see that, while FNN performs well in all the models, its performance is dependent on the complexity of the system dynamics. Generally, the larger the network size, the larger is the FNN; the lower the synchronization degree, the worse

are the predictions; and, comparing to phase oscillators, the inference of chaotic oscillators is more difficult. In addition to inferring the synchronization performance of relocated oscillators, the trained machine is also capable of inferring the synchronization performance of a new set of oscillators with different natural frequencies, showing preliminarily the feature of transfer learning (please see Supplemental Material [26] for the details). We finally note that the FNN technique is distinguished from the existing methods by its model-free feature, but not the computational efficiency, as the acquisition of the training data still relies on simulations. In addition, while evidence indicates that the machine has

learned successfully from data the mapping function between oscillator allocation and synchronization performance, the function is not explicitly given, i.e., the machine is working as a "black box."

[1] R. Albert and A.-L. Barabási, Statistical mechanics of complex networks, Rev. Mod. Phys. **74**, 47 (2002).

[2] J. A. Acebrón, L. L. Bonilla, C. J. Pérez Vicente, F. Ritort, and R. Spigler, The Kuramoto model: A simple paradigm for synchronization phenomena, Rev. Mod. Phys. **77**, 137 (2005).

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

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

[5] S. Boccaletti, J. A. Almendral, S. Guan, I. Leyva, Z. Liu, I. Sendiña-Nadal, Z. Wang, and Y. Zou, Explosive transitions in complex networks: structure and dynamics: Percolation and synchronization, Phys. Rep. **660**, 1 (2016).

[6] A. E. Motter and Z. Toroczkai, Introduction: Optimization in networks, Chaos **17**, 026101 (2007).

[7] D. J. Watts and S. H. Strogatz, Collective dynamics of 'small-world' networks, Nature (London) **393**, 440 (1998).

[8] M. Barahona and L. M. Pecora, Synchronization in Small-World Systems, Phys. Rev. Lett. **89**, 054101 (2002).

[9] A. E. Motter, C. Zhou, and J. Kurths, Weighted networks are more synchronizable: How and why, in *Science of Complex Networks: From Biology to the Internet and WWW: CNET 2004*, edited by J. F. F. Mendes, S. N. Dorogovtsev, A. Povolotsky, F. V. Abreu, and J. G. Oliveira, AIP Conf. Proc. Vol. 776 (AIP, Melville, NY, 2005), p. 201.

[10] X. G. Wang, Y.-C. Lai, and C.-H. Lai, Enhancing synchronization based on complex gradient networks, Phys. Rev. E **75**, 056205 (2007).

[11] T. Nishikawa, J. Sun, and A. E. Motter, Sensitive Dependence of Optimal Network Dynamics on Network Structure, Phys. Rev. X **7**, 041044 (2017).

[12] H. W. Fan, Y.-C. Lai, and X. G. Wang, Enhancing network synchronization by phase modulation, Phys. Rev. E **98**, 012212 (2018).

[13] W. Chen, J. Gao, Y. Lan, and J. Xiao, Optimizing synchrony with a minimal coupling strength of coupled phase oscillators on complex networks based on desynchronous clustering, Phys. Rev. E **105**, 044302 (2022).

[14] J. Gómez-Gardeñes, S. Gomez, A. Arenas, and Y. Moreno, Explosive Synchronization Transitions in Scale-Free Networks, Phys. Rev. Lett. **106**, 128701 (2011).

[15] M. Brede, Synchrony-optimized networks of non-identical Kuramoto oscillators, Phys. Lett. A **372**, 2618 (2008).

[16] Y. Wu, J. H. Xiao, G. Hu, and M. Zhan, Synchronizing large number of nonidentical oscillators with small coupling, Europhys. Lett. **97**, 40005 (2012).

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

[18] M. Zhan, S. Liu, and Z. He, Matching rules for collective behaviors on complex networks: Optimal configurations for vibration frequencies of networked harmonic oscillators, PLoS One **8**, e82161 (2013).

[19] P. S. Skardal, D. Taylor, and J. Sun, Optimal Synchronization of Complex Networks, Phys. Rev. Lett. **113**, 144101 (2014).

[20] T. F. Deng, W. Q. Liu, Y. Zhu, J. H. Xiao, and J. Kurths, Reviving oscillation with optimal spatial period of frequency distribution in coupled oscillators, Chaos **26**, 094813 (2016).

[21] P. S. Skardal, R. Sevilla-Escoboza, V. P. Vera-Ávila, and J. M. Buldú, Optimal phase synchronization in networks of phase-coherent chaotic oscillators, Chaos **27**, 013111 (2017).

[22] D. Svozil, V. Kvasnicka, and J. Pospichal, Introduction to multilayer feed-forward neural networks, Chemom. Intell. Lab. Syst. **39**, 43 (1997).

[23] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, Cambridge, MA, 2016).

[24] H. Zhao, Inferring the dynamics of "black-box" systems using a learning machine, Sci. China: Phys., Mech. Astron. **64**, 270511 (2021).

[25] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Learning representations by back-propagating errors, Nature (London) **323**, 533 (1986).

[26] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevE.107.024314 for more details about the FNN technique, the statistical properties of the data sets, the features of the optimal allocation scheme, and the results for other network models.

[27] A.-L. Barabási and R. Albert, Emergence of scaling in random networks, Science **286**, 509 (1999).

[28] L. M. Pecora, F. Sorrentino, A. M. Hagerstrom, T. E. Murphy, and R. Roy, Cluster synchronization and isolated desynchronization in complex networks with symmetries, Nat. Commun. **5**, 4079 (2014).

[29] C. Zhou, L. Zemanova, G. Zamora, C. C. Hilgetag, and J. Kurths, Hierarchical Organization Unveiled by Functional Connectivity in Complex Brain Networks, Phys. Rev. Lett. **97**, 238103 (2006).

[30] J. G. White, E. Southgate, J. N. Thomson, and S. Brenner, The structure of the nervous system of the nematode Caenorhabditis elegans, Philos. Trans. R. Soc. London, Ser. B **314**, 1 (1986).

[31] M. G. Rosenblum, A. S. Pikovsky, and J. Kurths, Phase Synchronization of Chaotic Oscillators, Phys. Rev. Lett. **76**, 1804 (1996).

[32] A. G. Bunn, D. L. Urban, and T. H. Keitt, Landscape connectivity: A conservation application of graph theory, J. Environ. Manage. **59**, 265 (2000).

[33] L. Stone and D. He, Chaotic oscillations and cycles in multi-trophic ecological systems, J. Theor. Biol. **248**, 382 (2007).