Chaos
An Interdisciplinary Journal of Nonlinear Science

Forecasting macroscopic dynamics in adaptive Kuramoto network using reservoir computing

Cite as: Chaos **32**, 103126 (2022); https://doi.org/10.1063/5.0114127
Submitted: 25 July 2022 • Accepted: 30 September 2022 • Published Online: 31 October 2022

Andrey V. Andreev, Artem A. Badarin, Vladimir A. Maximenko, et al.

Forecasting macroscopic dynamics in adaptive Kuramoto network using reservoir computing

Cite as: Chaos **32**, 103126 (2022); doi: 10.1063/5.0114127
Submitted: 25 July 2022 · Accepted: 30 September 2022 ·
Published Online: 31 October 2022

Andrey V. Andreev,[1,2] Artem A. Badarin,[1,3] Vladimir A. Maximenko,[1,2] and Alexander E. Hramov[1,2,a)]

AFFILIATIONS

[1]Engineering School of Information Technologies, Telecommunications and Control Systems, Ural Federal University named after the first President of Russia B.N.Yeltsin, 19 Mira str., 620002 Ekaterinburg, Russia
[2]Center for Neurotechnology and Machine Learning, Immanuel Kant Baltic Federal University, 14, A. Nevskogo str., Kaliningrad 236016, Russia
[3]Laboratory of Neuroscience and Cognitive Technology, Innopolis University, 1 Universitetskaya str., Kazan 240500, Russia

[a)]**Author to whom correspondence should be addressed:** hramovae@gmail.com

ABSTRACT

Forecasting a system's behavior is an essential task encountering the complex systems theory. Machine learning offers supervised algorithms, e.g., recurrent neural networks and reservoir computers that predict the behavior of model systems whose states consist of multidimensional time series. In real life, we often have limited information about the behavior of complex systems. The brightest example is the brain neural network described by the electroencephalogram. Forecasting the behavior of these systems is a more challenging task but provides a potential for real-life application. Here, we trained *reservoir computer* to predict the macroscopic signal produced by the network of phase oscillators. The Lyapunov analysis revealed the chaotic nature of the signal and *reservoir computer* failed to forecast it. Augmenting the feature space using Takkens' theorem improved the quality of forecasting. RC achieved the best prediction score when the number of signals coincided with the embedding dimension estimated via the nearest false neighbors method. We found that short-time prediction required a large number of features, while long-time prediction utilizes a limited number of features. These results refer to the bias-variance trade-off, an important concept in machine learning.

Machine learning (ML) is a state-of-the-art computational instrument that facilitates data analysis in various fields, including nonlinear dynamics, climate, and medicine. ML has gained popularity in neuroscience due to its ability to recognize hidden patterns and nonlinear relations in large sets of multi-modal neuroimaging data. The latest trends in medicine set tasks to predict the dynamics of neurophysiological systems. Prediction is the heart of prognostic medicine aimed at early-stage diagnosis and preventive influence on the human nervous system to preclude developing diseases. Recurrent neural networks (RNNs) offer a potential model for addressing these challenges. Reservoir Computing (RC), a type of RNN, handles this issue by mapping input signals into higher dimensional computational spaces through the dynamics of a fixed nonlinear system called a reservoir. Monitoring brain activity involves the registration of electroencephalogram (EEG) or magnetoencephalogram (MEG) signal. EEG/MEG refers to macroscopic signals measuring the mean electric field produced by thousands of neurons, but the contribution of each neuron is unknown. Forecasting the behavior of these systems is a more challenging task but provides a potential for real-life application. Here, we address this issue by using RC to forecast the macroscopic signal produced by the adaptive network of Kuramoto phase oscillators. We have shown that prediction requires a correct choice of the dimensionality of the reconstructed phase space of a macroscopic signal. To define dimensionality, we applied the nearest false neighbors method and estimated the embedding dimension. We have demonstrated that long-term predicting requires the number of reconstructed signals equal to or less than the embedding dimension. For the short-term prediction, the number of reconstructed signals must exceed the embedding dimension. We hypothesize that these results refer to the bias-variance trade-off important machine learning issue. With the high bias, the ML model misses the relevant relations between features and target outputs and fails to predict the system's behavior (underfitting). With the high variance, the ML algorithm models the random noise in the training

data and lacks generalization to unseen data (overfitting). We suppose that a growing number of features may increase variance due to the increasing complexity of the approximated function. For short-time prediction, it allows fitting the local patterns, but for long-time prediction, it reduces the algorithm's ability to fit more global trends.

I. INTRODUCTION

Machine learning (ML) provides powerful tools for forecasting the dynamics of complex systems of different natures like weather, climate,[1,2] medicine,[3,4] neuroscience,[5–7] ocean and turbulent streams,[8,9] traffic flows,[10,11] etc. In medicine, forecasting attracts interest due to the possibility of predictive intervention to preclude the development of the pathological activity. The brightest example is the prediction of epileptic.[12,13]

Signals received from real-world systems often exhibit chaotic dynamics that are difficult to predict. ML offers a promising approach to address these challenges known as reservoir computing (RC).[14–17] RC has shown significant success in modeling the full-order space dynamics of high-dimensional chaotic systems. Previously, RC has been successfully applied for predicting chaotic dynamics of finite-dimensional Rössler, Lorenz-63 systems, and spatially-distributed Kuramoto–Sivashinsky dynamical systems.[15,16,18] Pathak *et al.*[18] have shown that RC may modulate the system's dynamics: after correct short-term prediction, the actual and predicted trajectories start to scatter, but the predicted one continues to reproduce the system's attractor.

In real life, predicting the macroscopic dynamics of complex network systems is of the greatest importance. The problem is that signals from many separate objects through a complex communication structure merge into a single macroscopic signal, which significantly reduces the dimension of the system, making it impossible to restore the sources of a signal. In the bulk of real-life networks, the connections between elements change over time, making their macroscopic dynamics even more complex and unpredictable.

In a network of neurons in the brain, one has to distinguish between the signal registered at a microscopic level (the individual activity of a neuron) and the macroscopic signal, which is instead produced by a neuronal subnetwork.[19] Processes taking place at the microscopic level (for example, phase synchronization between neurons, formation of synchronous clusters, etc.) modulate the features of the macroscopic signal.[20,21] In the experiments, the signals recorded from the head's surface, e.g., electroencephalograms (EEGs), are macroscopic signals generated by a particular neuronal subnetwork. Prediction of these macroscopic signals is necessary for timely response to possible negative developments in their dynamics or for facilitating the brain–computer interaction.[22]

Here, we trained *reservoir computer* to forecast the macroscopic signal generated by the adaptive network and proposed an approach to increase the prediction quality. As a network model, we used Kuramoto oscillators connected with the links whose strength evolved under the homophilic and homeostatic principles. We showed that the Kuramoto network generated a chaotic signal, and *reservoir computer* failed to predict it. To improve the prediction quality, we reconstructed the phase space by adding the delayed signals (Takken's approach) and investigated how the prediction depends on a number of delays. We demonstrated that the optimal number of the delayed signals is two which corresponds to the embedding dimension. We also found that adding two delays provided an accurate long-term prediction, but the correct short-term prediction required more delays depending on the prediction horizon.

II. METHODS

A. Model of adaptive network of Kuramoto oscillators

We analyze the dynamics of a network of $N_K = 100$ Kuramoto phase oscillators analyzed in detail in Refs. 23–25. Each oscillator is described by the following equation:

$$\dot{\phi}_i(t) = \omega_i + \sum_{j \neq i} w_{ij}(t) \sin(\phi_j - \phi_i), \tag{1}$$

where $i = 1, \ldots, N_K$, $\{\omega_i\}$ is a set of randomly assigned natural frequencies distributed uniformly in $[-\pi, \pi]$ and w_{ij} is the weight of the connection between elements i and j. It is allowed to evolve in time according to the rule from Ref. 26. For each oscillator i and at each time t, the set of connection weights $\{w_{ij}\}$ satisfies the condition

$$\sum_{j \neq i}^{N_K} w_{ij} = 1. \tag{2}$$

The adaptive evolution of weights w_{ij} is governed by

$$\dot{w}_{ij}(t) = p_{ij}(t) - \left(\sum_{k \neq i} p_{ik}(t) \right) w_{ij}(t), \tag{3}$$

where the time dependent quantity $p_{ij}(t)$ is defined as

$$p_{ij}(t) = \frac{1}{T_m} \left| \int_{t-T_m}^{t} \exp^{i(\phi_i(t') - \phi_j(t'))} \, dt' \right|. \tag{4}$$

Here, $p_{ij}(t)$ denotes, at time t, the average phase correlation between oscillators i and j over a characteristic memory time $T_m = 100$. Equations (2) and (3) describe homeostatic and homophilic processes, respectively. So, this model describes the adaptive network of phase oscillators with the competition between homophily and homeostasis.

Initially, all the weights w_{ij} and phases ϕ_i are random. At time the couplings between the oscillators are changing along with the topology. As a result, a number of clusters can be formed because of the adaptation process: the connections between those units, which are phase correlated across memory time T_m, are getting stronger. So, Fig. 2(a) represents the resulted network with five clusters, inside each of them all phase oscillations are phase correlated. Following the paper (Ref. 19), we consider a macroscopic signal averaged over all $N_K = 100$ phase oscillators,

$$X_{avr}(t) = \frac{1}{N_K} \sum_{i=1}^{N_K} \sin[\phi_i(t)]. \tag{5}$$

To solve the differential equations, we use the Runge–Kutta fourth order method with time step $\Delta t = 0.1$ s for $T = 7000$ s.

For network structure visualization, we use ForceAtlas2 algorithm in the Gephi Software.[27] Based on the adjacency matrix the algorithm groups, the oscillators with high coupling strengths and distance are the ones with weak couplings. Due to the adaptation process, a coupling between two oscillators is strengthened if their phases are close to each other, so, inside each group, all oscillators are phase correlated.

B. Reservoir computing

We use a *reservoir computer* construct known as an echo state network, which uses a network of nodes as the internal reservoir.[18,28] The schematic structure of the network is shown in Fig. 1. The network has the input, hidden (reservoir), and output layers. Every reservoir node has inputs drawn from other nodes in the reservoir or the input to the RC, and every input has an associated weight. Each reservoir node also has an output, described by the following equation:

$$\mathbf{h}_t = \tanh(\mathbf{W}_{h,i}\mathbf{o}_t + \mathbf{W}_{h,h}\mathbf{h}_{t-1}), \tag{6}$$

where $\mathbf{h}_t$ is the internal high-dimensional hidden state, which enables the encoding of temporal dependencies on the past state history; $\mathbf{W}_{h,i}$ is the input-to-hidden $d_h \times d_o$ couplings matrix, which values are uniformly sampled from $[-\sigma_{\text{in}}, \sigma_{\text{in}}]$, where σ_{in} is the hyperparameter; d_h and d_o are the numbers of neurons in hidden and input layers, respectively; $\mathbf{W}_{h,h}$ is the reservoir (hidden-to-hidden) $d_h \times d_h$ matrix, which is set to a large low-degree matrix (the mean node degree D is the hyperparameter, the degrees of all nodes are randomly distributed with the mean value D), scaled appropriately to possess a spectral radius (absolute value of the largest eigenvalue) R whose value is also the hyperparameter; $\mathbf{o}_t$ is the d_h dimensional vector of the inputs. The output layer (d_o dimensional vector) is described by

$$\hat{\mathbf{o}}_{t+1} = \mathbf{W}_{o,h}\tilde{\mathbf{h}}_t, \tag{7}$$

where the augmented hidden state $\tilde{\mathbf{h}}_t$ is the d_h-dimensional vector such that the ith component of $\tilde{\mathbf{h}}_t$ is $\tilde{h}_t^i = h_t^i$ for half of the reservoir nodes and $\tilde{h}_t^i = (h_t^i)^2$ for the other half, enriching the dynamics with the square of the hidden state in half of the nodes; $\mathbf{W}_{o,h}$ is the hidden-to-output $d_o \times d_h$ matrix.

The output of each reservoir node is fed into the output layer of the RC, which performs a linear operation of the node values to produce the output of the RC as a whole.

During the training process, we apply a signal from the Kuramoto network $X_{avr}(t)$ to the input $\mathbf{o}_t$ of RC and receive output signal $\hat{\mathbf{o}}_{t+1}$. The goal of RC is to approximate the desired outputs $\hat{\mathbf{o}}_{t+1}$ appropriate to the inputs $\mathbf{o}_{t+1}$. To achieve it, for each $\mathbf{o}_t$, we use the Tikhonov regularized regression procedure[29] to alleviate overfitting by penalizing large values of the fitting parameters and find an

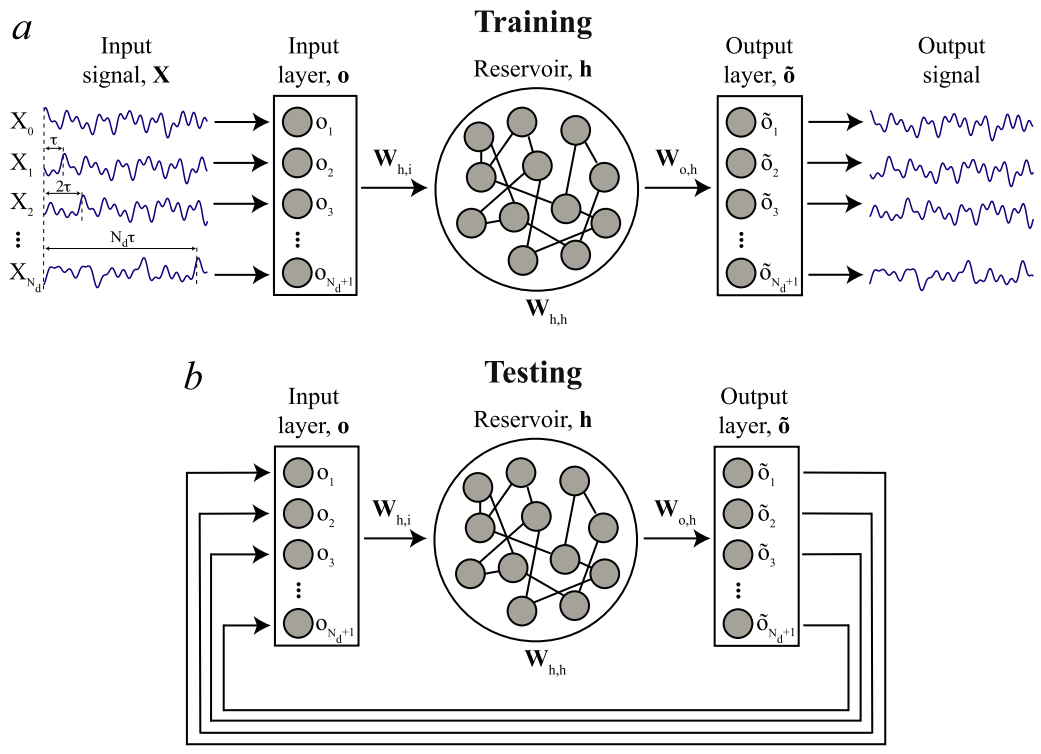

FIG. 1. Schematic presentation of the RC network in the training mode (a) and predicting (testing) mode (b). The number of input signals is equal to $(N_d + 1)$, where N_d is the number of the delay coordinates.

output matrix $\mathbf{W}_{o,h}$ that minimizes the following function:

$$\sum_{t=0}^{T_{\text{train}}} \left\| \mathbf{W}_{o,h}\tilde{\mathbf{h}}_t - \mathbf{o}_{t+1} \right\|^2 + \beta ||\mathbf{W}_{o,h}||^2, \tag{8}$$

where $||\mathbf{W}_{o,h}||^2$ is the sum of the squares of elements of $\mathbf{W}_{o,h}$ and $T_{\text{train}} = 5000$ is the time of the training process, $\beta = 10^{-8}$.

After training is complete, we start the testing process when RC tries to predict the signal dynamics by itself. We apply only the first point of the signal $X_{avr}(t)$ to the reservoir's input $\mathbf{o}_t$, after that the output $\hat{\mathbf{o}}_{t+1}$ is fed into the input $\mathbf{o}_{t+1}$, and the reservoir system is run autonomously.

We use the reservoir with $N_{\text{R}} = 1000$ nodes and investigate *reservoir computers* with a different number of input nodes. The number of output nodes is the same as the number of input nodes.

C. Methods for estimating networks dynamics

To estimate the effectiveness of the reservoir's prediction, we use the Pearson correlation and the normalized root mean square error. The Pearson linear correlation coefficient is described as follows:

$$r(o,\hat{o}) = \frac{\sum_{i=1}^{n}(o_i - \bar{o})(\hat{o}_i - \bar{\hat{o}})}{\sqrt{\sum_{i=1}^{n}(o_i - \bar{o})^2}\sqrt{\sum_{i=1}^{n}(\hat{o}_i - \bar{\hat{o}})^2}}, \tag{9}$$

where o is a target signal, $\hat{o}$ is a predicted signal, n is the signal size, $\bar{o} = \frac{1}{n}\sum_{i=1}^{n} o_i$ and analogously for $\bar{\hat{o}}$. $r = 1$ or -1 corresponds to perfect positive and negative correlation, respectively.

The Euclidean norm is described by the following equation:

$$\epsilon = ||o - \hat{o}|| = \sqrt{\sum_{i=1}^{n}(o_i - \hat{o}_i)^2}. \tag{10}$$

To estimate whether the investigated signal's dynamics are chaotic or not, we calculate the *largest Lyapunov exponent* Λ_1 from an experimental time series by the Wolf algorithm.[30,31] We take a starting point x_0 at the original time series, find another point x_0' close to x_0 by distance $||x_0' - x_0|| = \epsilon_0$ but not close at time, and consider two time series until the distance between x_1' and x_1 are greater than $\epsilon_{\max}$, fix time T_1 and ratio ϵ_0'/ϵ_0, find new point x_1'' close to x_1 by distance $||x_0'' - x_0|| = \epsilon_1$ and repeat the algorithm a multiple time, so, $\Lambda_1 = \left(\sum_{k=0}^{K-1}\ln(\epsilon_k'/\epsilon_k)\right) \Big/ \left(\sum_{k=1}^{K} T_k\right)$, where K is the number of algorithm repetitions.

We refer to the largest Lyapunov exponent as the maximal Lyapunov exponent (MLE) because it determines a notion of predictability for a dynamical system. Chaotic dynamics of a bounded trajectory is defined by the condition $\Lambda_1 > 0$. It determines the rate at which typical pairs of nearby orbits separate. To analyze the effectiveness of reservoir's prediction, we will use the Lyapunov time $T^{\Lambda_1} = 1/\Lambda_1$ instead of common time. It is defined as the time for the distance between nearby trajectories of the system to increase by a factor of e.

To estimate the embedding dimension of the analyzed system, we use False Nearest Neighbor (FNN) algorithm.[32] We evaluate the number of points being the false neighbors in the reconstructed phase space of dimension d. The point X_i and its nearest point X_i^* ($i = 1, \ldots, N_t$, where N_t is the number of time points in the signal) are false neighbors if

$$\sqrt{\frac{\epsilon_i^2(d+1) - \epsilon_i^2(d)}{\epsilon_i^2(d)}} < \delta, \tag{11}$$

where δ is the distance threshold and $\epsilon_i^2(d) = ||X_i - X_i^*||^2$ is the distance metric. One then repeats the process at higher dimensions d, stopping when the proportion of false nearest neighbors becomes zero or sufficiently small and will remain so from then onward. So, the estimated embedding dimension d is the smallest value that satisfies the condition $p_{\text{fnn}} < \xi$, where p_{fnn} is the ratio of FNN points to the total number of points in the reconstructed phase space, ξ is a small number.

To increase the dimension of the analyzed system to reconstruct the phase space, we add the delayed signals to the original one.[33]

To estimate the delay time, we use average mutual information (AMI) defined as

$$\text{AMI}(L) = \sum_{i=1}^{N} p(X_i, X_{i+L}) \log\left[\frac{p(X_i, X_{i+L})}{p(X_i)p(X_{i+L})}\right], \tag{12}$$

where N is the length of the time series, L is a delay, and $p(X)$ is a probability. The time delay τ is set to be the first local minimum of AMI.

III. RESULTS

We investigate the capability of RC to predict the dynamics of macroscopic signal [Eq. (5)] received from the network of 100 Kuramoto phase oscillators [Eq. (1)] with adaptive topology [Eqs. (2) and (3)]. The resulted signal [Eq. (5)] and its amplitude spectrum are shown in Figs. 2(b) and 2(c), respectively. Due to the adaptation process, the dynamics of the macroscopic signal is weakly non-stationary. We use one part of the signal ($0 \le t \le 5000$) in a training process and the other ($5000 < t \le 7000$) as a test sequence. To study how much the adaptation influences the network's dynamics, we calculate the Morlet-based wavelet spectrum[34] of the macroscopic signal [Fig. 2(d)]. As one can see, the macroscopic signal is characterized by four frequencies: three lowest ones remain the same for all time while the highest frequency periodically appears and disappears because of the adaptation.

To estimate whether the investigated macroscopic signal's dynamics is chaotic or not, we calculate the maximal Lyapunov exponent $\Lambda_1 = 0.8$, which defines the dynamics of the signal as chaotic (Sec. II C). The corresponding Lyapunov time for the analyzed system is $T^{\Lambda_1} = 1.25$ s.

The construction of *reservoir computer* is described in Sec. II B. For training process, we use the first part of the macroscopic signal for $0 \le t \le 5000$ [Fig. 2(a)]. After the training is complete, we apply the first point of the testing part of the macroscopic signal ($5000 < t \le 7000$) as an initial condition for RC and compare the actual signal and the predicted one. We repeat the described procedure for a set of reservoir hyperparameters: $2 \le D \le 10, 0.4 \le R \le 1.2, 0.001 \le \sigma_{\text{in}} \le 1.0$.

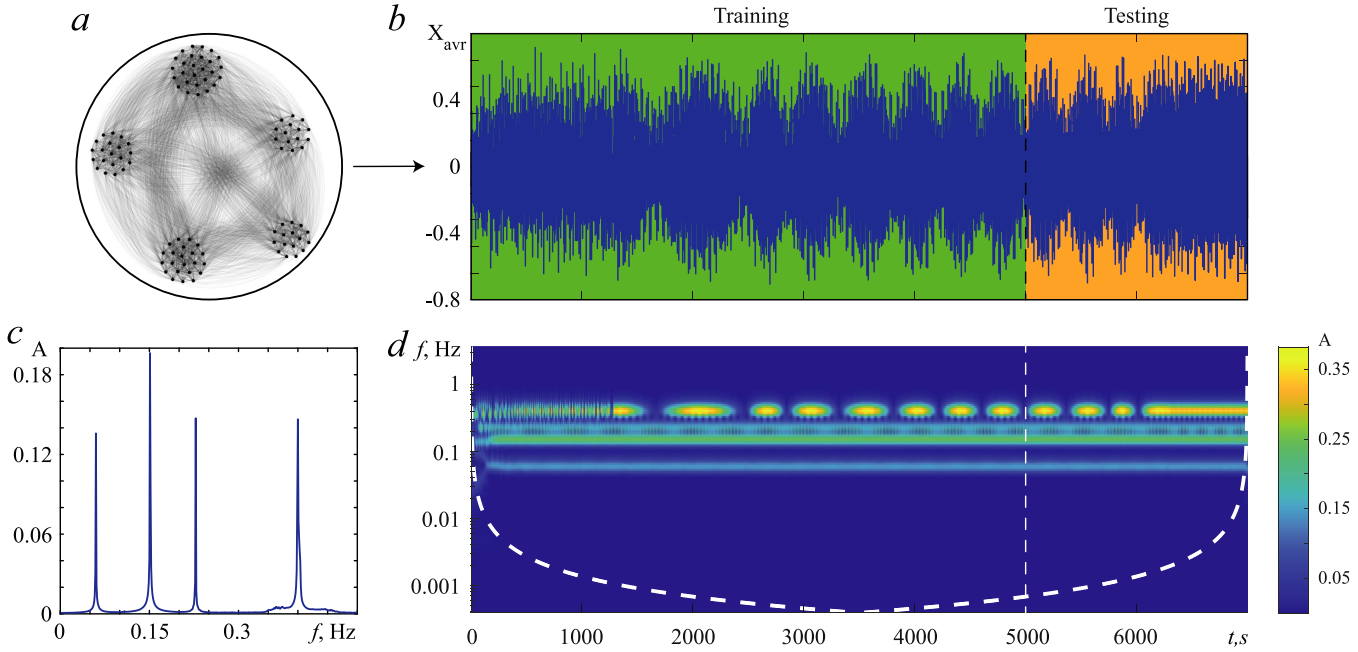

FIG. 2. (a) Schematic presentation of the topology of 100 Kuramoto phase oscillators network with the adaptation of the couplings. (b) The macroscopic signal [Eq. (5)] obtained from the Kuramoto network by averaging the dynamics of all network's elements. One part of the signal is used in the training process and the other in the predicting process. (c) Fourier and (d) wavelet spectra of the considered macroscopic signal. The vertical dashed curves in (b) and (d) separate the training and testing parts of the signal. The horizontal dashed line in (d) separates regions below it where edge effects are significant.

To find the optimal triplet of hyperparameters $(D, R, \sigma_{\text{in}})$, we calculate Pearson's linear correlation coefficient r [Eq. (9)] for the actual and predicted signals for $25T^{\Lambda_1}$ Lyapunov times. The maximal correlation $r = 0.863$ is achieved for $D = 9, R = 1.2, \sigma_{\text{in}} = 0.3$. The prediction for time interval $t \in [0, 125T^{\Lambda_1}]$ is shown in Fig. 3(a) for the optimal set of reservoir hyperparameters. As one can see, the quality of such prediction is low, only the first oscillation is well predicted, the next seven reservoir's oscillations coincide with the actual ones by phase and some of them are similar by amplitude. After 25 Lyapunov times, the predicted and actual trajectories become almost uncorrelated.

To compare the actual and predicted dynamics, we calculate amplitude spectra for them [Fig. 3(b)]. One can see, that RC learns to reproduce only one the highest frequency of 0.4 Hz. It also shows

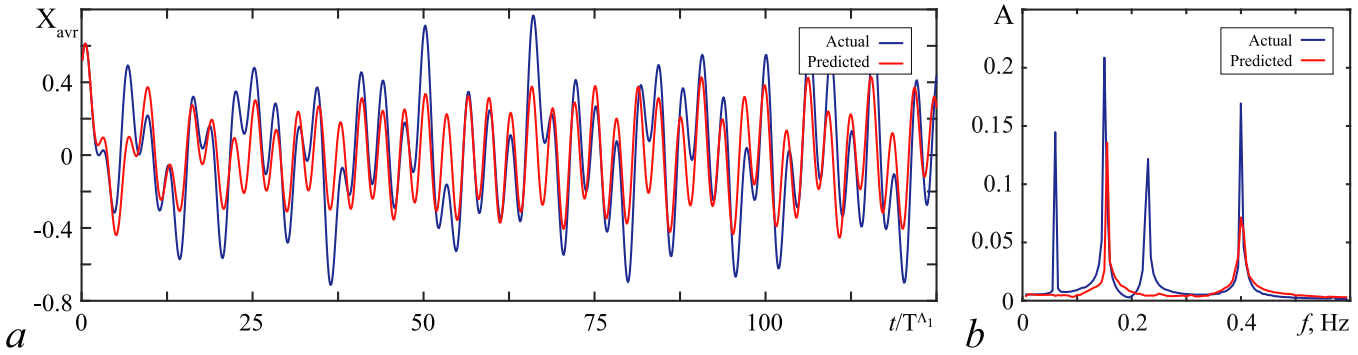

FIG. 3. (a) The state prediction (red) of the reservoir and the actual trajectory (blue) of the macroscopic signal of the adaptive Kuramoto network and (b) the corresponding amplitude spectra for the actual and predicting macroscopic signals for the case when we use only the initial signal $X_{avr}(t)$ as the reservoir input. Hyperparameters of RC is $D = 9, R = 1.2, \sigma_{\text{in}} = 0.3$. $T^{\Lambda_1} = 1.25$ s.

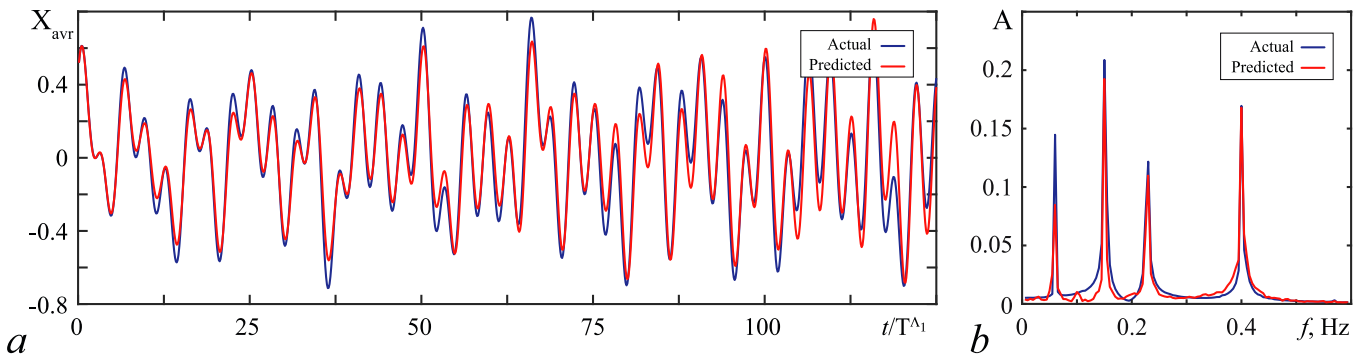

FIG. 4. (a) The state prediction (red) of the reservoir and the actual trajectory (blue) of the macroscopic signal of the adaptive Kuramoto network and (b) the corresponding amplitude spectra for the actual and predicting macroscopic signals for the case when we add $N_d = 2$ delayed signals to the original one to the reservoir input. Hyperparameters of RC is $D = 2, R = 0.8, \sigma_{\text{in}} = 0.1$. $T^{\Lambda_1} = 1.25$ s.

oscillations at the frequency close to the most powerful spectral component of 0.15 Hz and does not reproduce the other two frequencies 0.07 and 0.23 Hz.

To improve the quality of prediction, we reconstruct the phase space of the macroscopic signal by adding the delayed signals to the original one:[33] $\mathbf{X} = \{X_{avr}(t), X_{avr}(t-\tau), X_{avr}(t-2\tau), \ldots, X_{avr}(t-N_d\tau)\}^T$, where N_d is the number of delay coordinates.

We estimate the embedding dimension of the considered signal by using FNN algorithm[32] and find that the embedding dimension is 3. So, we add $N_d = 2$ delayed signals to the original one with time delay $\tau = 4.5$ s and apply three signals $\{X_{avr}(t), X_{avr}(t-\tau), X_{avr}(t-2\tau)\}$ to the input of RC. The time delay $\tau = 4.5$ s was defined by using Average Mutual Information [Eq. (12)].

Similar to the case with one input signal, we find the optimal triplet of hyperparameters. The maximal correlation $r = 0.993$ for $25T^{\Lambda_1}$ Lyapunov times is achieved for $D = 2, R = 0.8, \sigma_{\text{in}} = 0.1$. The prediction is shown in Fig. 4(a) for the obtained optimal set of hyperparameters. For 50 Lyapunov times, the predicted signal demonstrates very similar dynamics to the original one. All phases of the signals during this time are the same, amplitudes of original and predicted signals are close to each other. Amplitude spectra of the signals are shown in Fig. 4(b). As one can see, adding two delayed signals to RC's input improves the quality of prediction, RC reproduces all four frequencies of the original signal with high accuracy.

We investigate the sensitivity of the reported results to a particular trajectory (initial conditions and time step). We choose $N_d = 2, D = 2, R = 0.8, \sigma_{\text{in}} = 0.1$ and change the initial conditions of the testing sequence in a range of the signal oscillations. As a result, the correlation $r = 0.976 \pm 1.76\%$ for $25T^{\Lambda_1}$ Lyapunov times changes in a small range from 0.959 to 0.993 with mean value 0.976. Increasing the time step of the signal (both training and testing parts) by 2, 3, 4, and 5 times reduces the correlation by 1.2%, 4.7%, 7.3%, and 10.4%, respectively.

The delay time $\tau = 4.5$ s for all calculations presented. Figure 5(a) illustrates the maximum achievable correlations vs the number of delay coordinates N_d. The blue line corresponds to the correlation calculated during 25 Lyapunov times, the red one—during 125 Lyapunov times. It is clearly seen that adding one delayed signal does not increase the correlation sufficiently, but adding two signals highly increases both correlations. Further increasing the number of input signals $(N_d + 1)$ only slightly increases r_{max} during time interval $125T^{\Lambda_1}$.

We also calculate the maximal time during which the correlation r is more than 0.8 for a different number of the delay

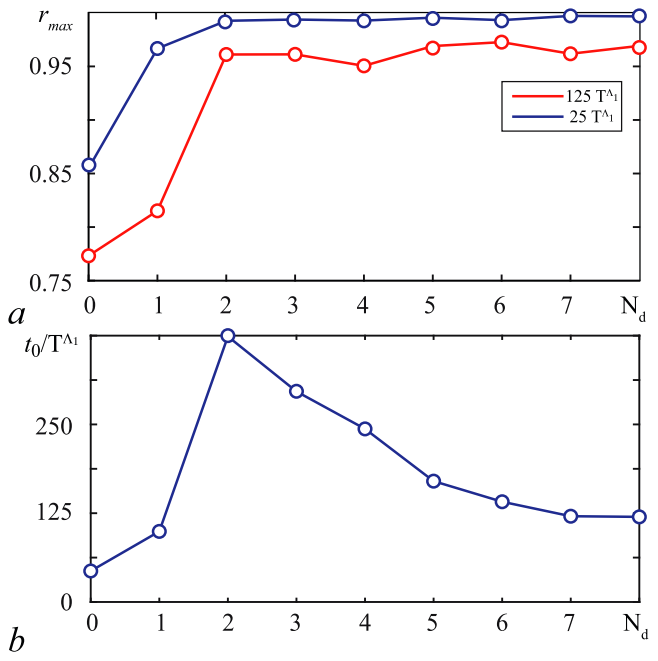

FIG. 5. (a) The maximal correlation r_{max} between the actual and predicted macroscopic signals during 25 (black) and 125 (red) Lyapunov times T^{Λ_1} vs the number of the delayed coordinates N_d. (b) The dependence of the time interval t_0 during which the correlation $r > 0.8$ on the number of delay coordinates N_d. $T^{\Lambda_1} = 1.25$ s.

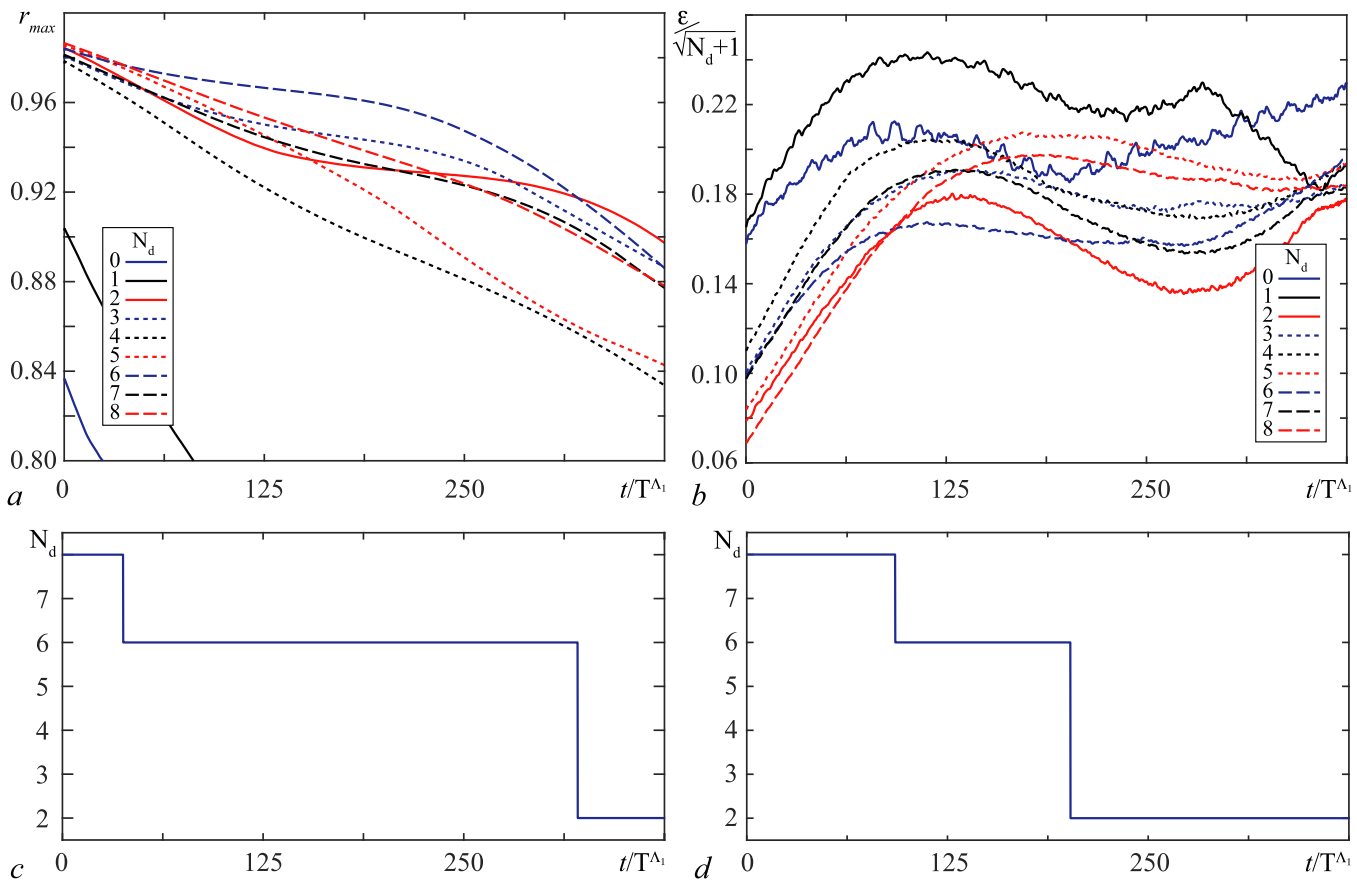

FIG. 6. Time dependencies of (a) the maximal correlation r_{max} and (b) the normalized error $\epsilon/\sqrt{N_d+1}$ [Eq. (10)] between the actual and predicted macroscopic signals and among all the considered parameters (D, R, σ_{in}) for different number of delayed coordinates $N_d \in [0, 8]$. (c) The number of delay coordinates N_d for which correlation in Fig. (a) is maximal. (d) The number of delay coordinates N_d for which $\epsilon/\sqrt{N_d+1}$ in Fig. (b) is minimal.

coordinates N_d [Fig. 5(b)]. We find that the longest time during which $r_{max} > 0.8$ is equal to $375T^{\Lambda_1}$ and corresponds to three input signals ($N_d = 2$). This result correlates with the estimated embedding dimension of the original signal, which is equal to 3. Further increasing the number of delayed signals leads to decreasing the prediction time. So, adding $N_d = 2$ delays is the most universal solution for the task of dynamics prediction for the investigated Kuramoto network system because it allows us to predict its macroscopic dynamics correctly for the longest time.

Finally, we study how the optimal number N_d of delays depends on the time we need to predict. To do this, we fix time interval and calculate correlations between the actual and predicted signals for the time over all parameters. Then, we choose the maximal correlation r_{max}, increase the time, and repeat. The resulting dependencies are shown in Fig. 6(a) for different values of N_d. Based on these results, we choose the number of delays N_d for which the correlation is maximal for the time we need to predict for [Fig. 6(c)]. As one can see, for small time ($t < 38T^{\Lambda_1}$), the best prediction is achieved when we use $N_d = 8$ delay coordinates. For bigger time ($38 < t/T^{\Lambda_1} < 320$), we need to decrease N_d to 6 for the best accuracy. If the prediction time is bigger, then $320T^{\Lambda_1}$ the number of delays should be decreased to 2. So, the more time we need to predict, the less input signals for reservoir we need to use to maintain the best prediction quality.

Based on the results above, we make an assumption that increasing the number of delays N_d has a positive and a negative impact: more delays give more information about the signal to the reservoir but contribute to a faster increase in error during the iterative process. To prove it, we calculate an error ϵ [Eq. (10)] as the norm of the difference between actual and predicted signals on each iteration for the different values of N_d. Figure 6(b) demonstrates the corresponding dependences of error ϵ on time t. For small prediction times ($t < 100T^{\Lambda_1}$), the error is small, and the positive effect of a large number of input signals prevails. As the prediction time increases, the error ϵ increases during the iterative process in the reservoir, and input information from large inputs ($N_d + 1$) stops to dominate over error. It leads to decreasing the optimal number of the used delays N_d. As a result, Fig. 6(d) shows the same tendency

as illustrated in Fig. 6(c): the more time we need to predict, the less input signals we need to use for achieving better results. Notably, the optimal number of delays for both (c) and (d) are only 8, 6, and 2.

IV. CONCLUSION

We investigated the capability of reservoir computing to predict the macroscopic signal generated by the adaptive network. As an analyzed model, we used the network of Kuramoto phase oscillators where the coupling between nodes changed over time according to adaptive rules. Using Lyapunov analysis, we confirmed the chaotic nature of the generated macroscopic signal.

We demonstrated that the reservoir trained on the raw macroscopic signal failed to predict it. To improve the prediction quality, we reconstructed the phase space of the macroscopic signal using Takken's approach, i.e., by adding the delayed signals. Using these delayed signals as the reservoir input increased the accuracy of the prediction.

We studied how the number of delayed signals influenced the quality and time horizon of the prediction. We found that the correlation between the original and the predicted signal peaked for two delays and remained unchanged with a further increase in the delays. We defined the prediction horizon as a time interval for which the correlation exceeded 0.8. The maximum horizon of prediction was also achieved for two delays. The observed optimal number of delays correlated with the embedding dimension of the original signal estimated via the nearest false neighbors method.

We found that the optimal number of delays depended on the prediction horizon: the long-term prediction required fewer input signals and vice versa. The potential reason is that increasing the number of delays has a positive and a negative impact: more delays give more information about the macroscopic signal to the reservoir but contribute to a faster increase in error during the iterative process.

We hypothesize that these results refer to the bias-variance trade-off important machine learning issue. With the high bias, the ML model misses the relevant relations between features and target outputs and fails to predict the system's behavior (underfitting). With the high variance, the ML algorithm models the random noise in the training data and lacks generalization to unseen data (overfitting). We suppose that a growing number of features may increase variance due to the increasing complexity of the approximated function. For short-time prediction, it allows fitting the local patterns, but for long-time prediction, it reduces the algorithm's ability to fit more global trends.

We believe that our results will be useful in the application of RC for the signals obtained from the real systems when the data are not enough for the sufficient training of a neural network. The approach proposed in the current paper could be combined with other methods to work with specific data. For example, while working with a multistable system, one needs to combine our approach with "the blending technique" explained in Refs. 35 and 36.

ACKNOWLEDGMENTS

The research funding from the Ministry of Science and Higher Education of the Russian Federation (Ural Federal University Program of Development within the Priority-2030 Program) is gratefully acknowledged.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Andrey V. Andreev: Formal analysis (equal); Investigation (equal); Software (equal); Visualization (equal); Writing – original draft (equal). **Artem A. Badarin:** Data curation (equal); Formal analysis (equal); Investigation (equal); Validation (equal). **Vladimir A. Maximenko:** Conceptualization (equal); Data curation (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Validation (equal). **Alexander E. Hramov:** Conceptualization (equal); Formal analysis (equal); Funding acquisition (equal); Methodology (equal); Project administration (equal); Resources (equal); Supervision (equal); Writing – review & editing (equal).

DATA AVAILABILITY

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

REFERENCES

[1]T. Gneiting and A. E. Raftery, "Weather forecasting with ensemble methods," Science **310**(5746), 248–249 (2005).
[2]J. A. Weyn, D. R. Durran, and R. Caruana, "Can machines learn to predict weather? Using deep learning to predict gridded 500-hPa geopotential height from historical weather data," J. Adv. Model. Earth Syst. **11**(8), 2680–2693 (2019).
[3]A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," Nature **542**(7639), 115–118 (2017).
[4]T. Kurth, S. Treichler, J. Romero, M. Mudigonda, N. Luehr, E. Phillips, A. Mahesh, M. Matheson, J. Deslippe, M. Fatica *et al.*,"Exascale deep learning for climate analytics," in *SC18: International Conference for High Performance Computing, Networking, Storage and Analysis* (IEEE, 2018), pp. 649–660.
[5]J. Richiardi, S. Achard, H. Bunke, and D. Van De Ville, "Machine learning with brain graphs: Predictive modeling approaches for functional imaging in systems neuroscience," IEEE Signal Process. Mag. **30**(3), 58–70 (2013).
[6]J. I. Glaser, A. S. Benjamin, R. Farhoodi, and K. P. Kording, "The roles of supervised machine learning in systems neuroscience," Progress Neurobiol. **175**, 126–137 (2019).
[7]D. Bzdok and J. P. Ioannidis, "Exploration, inference, and prediction in neuroscience and biomedicine," Trends Neurosci. **42**(4), 251–262 (2019).
[8]R. Lou, Z. Lv, S. Dang, T. Su, and X. Li, "Application of machine learning in ocean data," Multi. Syst. 1–10 (2021), available at https://link.springer.com/article/10.1007/s00530-020-00733-x.
[9]S. L. Brunton, B. R. Noack, and P. Koumoutsakos, "Machine learning for fluid mechanics," Annu. Rev. Fluid Mech. **52**, 477–508 (2020).
[10]J. Wojtusiak, T. Warden, and O. Herzog, "Machine learning in agent-based stochastic simulation: Inferential theory and evaluation in transportation logistics," Comput. Math. Appl. **64**(12), 3658–3665 (2012).
[11]N. Servos, X. Liu, M. Teucke, and M. Freitag, "Travel time prediction in a multimodal freight transport relation using machine learning algorithms," Logistics **4**(1), 1 (2020).
[12]N. S. Frolov, V. V. Grubov, V. A. Maksimenko, A. Lüttjohann, V. V. Makarov, A. N. Pavlov, E. Sitnikova, A. N. Pisarchik, J. Kurths, and A. E. Hramov, "Statistical properties and predictability of extreme epileptic events," Sci. Rep. **9**(1), 1–8 (2019).

[13]V. Grubov, E. Sitnikova, A. Pavlov, A. Koronovskii, and A. Hramov, "Recognizing of stereotypic patterns in epileptic EEG using empirical modes and wavelets," Phys. A **486**, 206–217 (2017).
[14]P. R. Vlachas, W. Byeon, Z. Y. Wan, T. P. Sapsis, and P. Koumoutsakos, "Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks," Proc. R. Soc. A **474**(2213), 20170844 (2018).
[15]J. Pathak, A. Wikner, R. Fussell, S. Chandra, B. R. Hunt, M. Girvan, and E. Ott, "Hybrid forecasting of chaotic processes: Using machine learning in conjunction with a knowledge-based model," Chaos **28**(4), 041101 (2018).
[16]A. Griffith, A. Pomerance, and D. J. Gauthier, "Forecasting chaotic systems with very low connectivity reservoir computers," Chaos **29**(12), 123108 (2019).
[17]K. Nakai and Y. Saiki, "Machine-learning inference of fluid variables from data using reservoir computing," Phys. Rev. E **98**(2), 023111 (2018).
[18]J. Pathak, Z. Lu, B. R. Hunt, M. Girvan, and E. Ott, "Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data," Chaos **27**(12), 121102 (2017).
[19]V. A. Maksimenko, A. Lüttjohann, V. V. Makarov, M. V. Goremyko, A. A. Koronovskii, V. Nedaivozov, A. E. Runnova, G. van Luijtelaar, A. E. Hramov, and S. Boccaletti, "Macroscopic and microscopic spectral properties of brain networks during local and global synchronization," Phys. Rev. E **96**(1), 012316 (2017).
[20]H.-J. Park and K. Friston, "Structural and functional brain networks: From connections to cognition," Science **342**(6158), 1238411 (2013).
[21]A. E. Hramov, N. S. Frolov, V. A. Maksimenko, S. A. Kurkin, V. B. Kazantsev, and A. N. Pisarchik, "Functional networks of the brain: From connectivity restoration to dynamic integration," Phys.-Usp. **64**(6), 584 (2021).
[22]A. E. Hramov, V. A. Maksimenko, and A. N. Pisarchik, "Physical principles of brain–computer interfaces and their applications for rehabilitation, robotics and control of human brain states," Phys. Rep. **918**, 1–133 (2021).
[23]F. A. Rodrigues, T. K. D. Peron, P. Ji, and J. Kurths, "The Kuramoto model in complex networks," Phys. Rep. **610**, 1–98 (2016).
[24]N. Frolov and A. Hramov, "Extreme synchronization events in a Kuramoto model: The interplay between resource constraints and explosive transitions," Chaos **31**(6), 063103 (2021).
[25]N. Frolov, V. Maksimenko, S. Majhi, S. Rakshit, D. Ghosh, and A. Hramov, "Chimera-like behavior in a heterogeneous Kuramoto model: The interplay between attractive and repulsive coupling," Chaos **30**(8), 081102 (2020).
[26]V. Makarov, A. Koronovskii, V. Maksimenko, A. Hramov, O. Moskalenko, J. M. Buldu, and S. Boccaletti, "Emergence of a multilayer structure in adaptive networks of phase oscillators," Chaos Solitons Fractals **84**, 23–30 (2016).
[27]M. Jacomy, T. Venturini, S. Heymann, and M. Bastian, "ForceAtlas2, a continuous graph layout algorithm for handy network visualization designed for the Gephi software," PLoS One **9**(6), e98679 (2014).
[28]P. R. Vlachas, J. Pathak, B. R. Hunt, T. P. Sapsis, M. Girvan, E. Ott, and P. Koumoutsakos, "Backpropagation algorithms and reservoir computing in recurrent neural networks for the forecasting of complex spatiotemporal dynamics," Neural Netw. **126**, 191–217 (2020).
[29]A. N. Tikhonov and V. Y. Arsenin, "Solutions of ill-posed problems," SIAM Rev. **1**(30), 487 (1977).
[30]A. Wolf, J. B. Swift, H. L. Swinney, and J. A. Vastano, "Determining Lyapunov exponents from a time series," Phys. D **16**(3), 285–317 (1985).
[31]J.-P. Eckmann, S. O. Kamphorst, D. Ruelle, and S. Ciliberto, "Liapunov exponents from time series," Phys. Rev. A **34**(6), 4971 (1986).
[32]C. Rhodes and M. Morari, "False-nearest-neighbors algorithm and noise-corrupted time series," Phys. Rev. E **55**(5), 6162 (1997).
[33]N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw, "Geometry from a time series," Phys. Rev. Lett. **45**(9), 712 (1980).
[34]A. E. Hramov, A. A. Koronovskii, V. A. Makarov, A. N. Pavlov, and E. Sitnikova, *Wavelets in Neuroscience* (Springer, 2015).
[35]A. Flynn, V. A. Tsachouridis, and A. Amann, "Multifunctionality in a reservoir computer," Chaos **31**(1), 013125 (2021).
[36]A. Flynn, J. Herteux, V. A. Tsachouridis, C. Räth, and A. Amann, "Symmetry kills the square in a multifunctional reservoir computer," Chaos **31**(7), 073122 (2021).