Transfer learning for nonlinear dynamics and its application to fluid turbulence

Masanobu Inubushi* and Susumu Goto
Graduate School of Engineering Science, Osaka University, 1-3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan

(Received 6 March 2020; revised 11 August 2020; accepted 8 September 2020; published 2 October 2020)

We introduce transfer learning for nonlinear dynamics, which enables efficient predictions of chaotic dynamics by utilizing a small amount of data. For the Lorenz chaos, by optimizing the transfer rate, we accomplish more accurate inference than the conventional method by an order of magnitude. Moreover, a surprisingly small amount of learning is enough to infer the energy dissipation rate of the Navier-Stokes turbulence because we can, thanks to the small-scale universality of turbulence, transfer a large amount of the knowledge learned from turbulence data at lower Reynolds number.

DOI: 10.1103/PhysRevE.102.043301

I. INTRODUCTION

Machine learning (ML) is becoming a powerful tool for a broad range of problems in physics, and it is likely to solve certain long-standing problems in nonlinear physics, such as turbulence modeling [1–3], in the near future. Solving these problems is not only crucial in fundamental physics, but also has an immeasurable impact upon practical problems, e.g., in fields such as mechanical engineering and weather forecasting.

As an ML method suitable for nonlinear dynamics, we focus on *reservoir computing (RC)* [4]. RC has been successfully applied to the problems of nonlinear dynamics such as the inference of unmeasured variables and the prediction of future states of spatiotemporal chaos [5–10]. Similarly to other ML methods, RC requires a large amount of training data. However, this requirement is often unsatisfied, i.e., the amount of training data is often limited. This fundamental problem would be a major bottleneck of making RC applicable, especially for spatiotemporal chaos with a large degree of freedom.

The ML-based turbulence modeling is a typical example. Although the ML-based turbulence modeling will be useful in practical engineering applications, these modeling requires a large amount of high-Reynolds-number turbulence data collected over a long period of time. The length of the training data required for RC exceeds the turnover time of the largest eddies by several thousand times, for which we give a theoretical estimation in Appendix A. Generating such turbulence data for practical applications, e.g., using the direct numerical simulation, is usually impossible. Therefore, it is essential to learn the knowledge of the high-Reynolds-number turbulence from a small amount of data.

To solve the fundamental problem, we employ *transfer learning*, which is a concept to utilize knowledge learned in a task for another different but similar task. Although transfer learning has been successfully used for ML tasks such as image classifications [11], it is unclear how useful this concept is and how to implement it for problems in physics.

In this paper, we develop a transfer learning method for RC [12] with an optimization of *transfer rate* (defined below), and then, we show that the method is indeed effective for tasks of chaotic dynamics. Taking the Lorenz equations as an example, we demonstrate that optimizing the transfer rate is essential, which leads to more accurate inference than the conventional method by an order of magnitude.

More importantly, concerning the inference of the energy dissipation rate of the Navier-Stokes turbulence, we uncover that the amount of learning can be drastically reduced, by transferring the knowledge learned from turbulence data at lower Reynolds number. A main conclusion is that the universality of the energy cascade of turbulence enables us to use a large transfer rate, which is crucial for the ML-based turbulence modeling.

II. METHOD

We study a dynamical system $\boldsymbol{x}(k+1) = \boldsymbol{f}_\rho[\boldsymbol{x}(k)]$ with some control parameter ρ and the inference task, although our proposed method is not limited to this task. The goal of the task is to infer an unmeasurable quantity $v[\boldsymbol{x}(k)]$ from a measurable quantity $u[\boldsymbol{x}(k)]$ at a parameter denoted by $\rho = \rho_\mathcal{T}$. The training data $\mathcal{D}_\mathcal{T}$ consists of the input data $u(k) = u[\boldsymbol{x}(k)]$ and the output data $v(k) = v[\boldsymbol{x}(k)]$, i.e., $\mathcal{D}_\mathcal{T} = \{u(k), v(k)\}_{1\leqslant k\leqslant T_L'}$. We consider that the length of the training data T_L' is not sufficiently long.

Here we assume that a sufficient amount of training data $\mathcal{D}_\mathcal{S} = \{u(k), v(k)\}_{1\leqslant k\leqslant T_L}$ is available at a parameter $\rho = \rho_\mathcal{S}$ which differs from the parameter $\rho_\mathcal{T}$, and these training data $\mathcal{D}_\mathcal{T}$ and $\mathcal{D}_\mathcal{S}$ are similar to each other. We refer to the parameter $\rho_\mathcal{S}$ and $\rho_\mathcal{T}$ as the *source* and *target* domains (parameters), respectively. Our method utilizes knowledge learned from the source domain to realize the inference in the target domain (see Fig. 1).

To derive explicit formulas, we use the echo state network (ESN) introduced by Jaeger (2001) [13] as a conventional RC method. The state variable r_i of the ith node in the ESN evolves in time as follows: $r_i(k+1) = \phi[\sum_{j=1}^N J_{ij} r_j(k) + \epsilon u(k) + \eta\xi_i]$, where N is the number of nodes, and J_{ij} and ξ_i are the fixed random connections and biases, respectively.

*inubushi@me.es.osaka-u.ac.jp

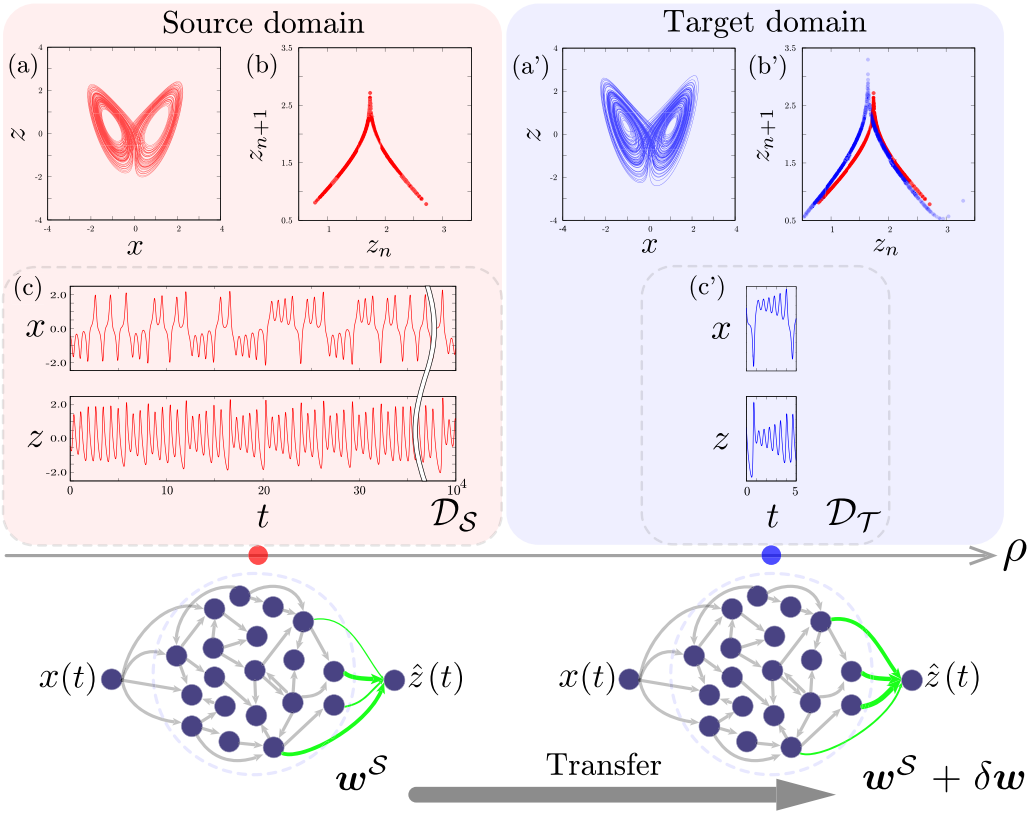

FIG. 1. Illustrative example of transfer learning for RC applied to the inference problem of $z(t)$ from $x(t)$ of the Lorenz chaos. The red-shaded region on the left shows the data of the Lorenz chaos in the source domain ($\rho_S = 28$). The blue-shaded region on the right shows the data in the target domain ($\rho_T = 40$). (a, a′) Projection of the attractor onto the $x - z$ plane. (b, b′) Lorenz plot. The data plotted in (b) is replotted in (b′) for comparison. (c, c′) Training data in $\mathcal{D}_S$ and $\mathcal{D}_T$. We consider the case that the training data in $\mathcal{D}_T$ is limited.

Further, $\phi[\cdot]$ is the so-called activation function, and we employ $\phi[x] = \tanh[gx]$. Here $g, \epsilon, \eta \in \mathbb{R}$ are hyperparameters. We used $N = 100$ nodes of ESN with the following hyperparameters: $g = 0.95$, $\epsilon = 0.2$, and $\eta = 0.01$. Elements J_{ij} of the connection matrix and the bias terms ξ_i are independently and identically drawn from the Gaussian distribution: $J_{ij} \sim \mathcal{N}(0, 1/N)$ and $\xi_i \sim \mathcal{N}(0, 1)$. The time-series $\{u(k), v(k)\}$ are normalized to have zero mean and unit variance.

Our method consists of the following two steps: (I) training in the source domain with $\mathcal{D}_S$, and (II) training in the target domain with $\mathcal{D}_T$.

(I) Training in the source domain: The readout from the ESN is given by $\hat{v}(k) = \sum_{i=1}^N w_i^S r_i(k)$. The hat symbol indicates the inferred quantities. The readout weight $\boldsymbol{w}^S$ is determined with $\mathcal{D}_S$ to minimize the mean-square error (MSE) $E(\boldsymbol{w}) = \langle (v - \hat{v})^2 \rangle_{T_L} = \langle (v - \sum_{i=1}^N w_i r_i)^2 \rangle_{T_L}$, where $\langle a \rangle_T := \frac{1}{T} \sum_{k=1}^T a(k)$. Calculating $\frac{\partial}{\partial w_j} E(\boldsymbol{w}) = 0\,(j = 1, \cdots, N)$, the trained readout weight is given by

$$\boldsymbol{w}^S = R^{-1}\boldsymbol{q}, \tag{1}$$

where $R_{ij} := \langle r_i r_j \rangle_{T_L}$ and $q_i := \langle v r_i \rangle_{T_L}$.

(II) Training in the target domain: We use the same ESN as in the source domain, and train the readout weight $\boldsymbol{w}^T$. Because of the similarity of the training data $\mathcal{D}_S$ and $\mathcal{D}_T$, the relation $\boldsymbol{w}^T \simeq \boldsymbol{w}^S + \delta\boldsymbol{w}$ would hold with a small correction $\delta\boldsymbol{w}$. Thus, we consider the readout from the ESN as $\hat{v}(k) = \sum_{i=1}^N (w_i^S + \delta w_i) r_i(k)$ with the weight $\boldsymbol{w}^S$ already trained by Eq. (1). The correction weight $\delta\boldsymbol{w}$ is determined to minimize the following function:

$$\mathcal{E}(\delta\boldsymbol{w}) = \left\langle \left(v'(t) - \sum_{i=1}^N (w_i^S + \delta w_i) r_i'(t) \right)^2 \right\rangle_{T_L'} + \mu \|\delta\boldsymbol{w}\|_2^2, \tag{2}$$

where $\|\delta\boldsymbol{w}\|_2^2 = \sum_{i=1}^N \delta w_i^2$. The variables with primes, such as v', denote variables in the target domain. We refer to the parameter $\mu \in [0, \infty]$ as the *transfer rate*. Calculating $\frac{\partial}{\partial \delta w_j} \mathcal{E}(\delta\boldsymbol{w}) = 0\,(j = 1, \cdots, N)$, the correction weight is given by

$$\delta\boldsymbol{w} = [R' + \mu I]^{-1}\boldsymbol{q}', \tag{3}$$

where $R'_{ij} := \langle r_i' r_j' \rangle_{T_L'}$, I is the identity matrix and $\boldsymbol{q}' := \langle v'\boldsymbol{r}' \rangle_{T_L'} - R'\boldsymbol{w}^S$.

The transfer rate μ, which is similar to l_2 regularization, controls the amount of knowledge transferred from the source domain to the target domain. If the transfer rate is zero, $\mu = 0$, then the above formula is reduced to the conventional RC method which is supervised learning by using the target data $\mathcal{D}_T$ only, i.e., transferring no knowledge from the source domain. In the limit of the large transfer rate, $\mu \to \infty$, we have $\|\delta\boldsymbol{w}\|_2 \to 0$ (see Appendix B for proof). Namely, in this limit, the above formula is reduced to a method that simply reuses the weight $\boldsymbol{w}^S$, i.e., no learning in the target domain.

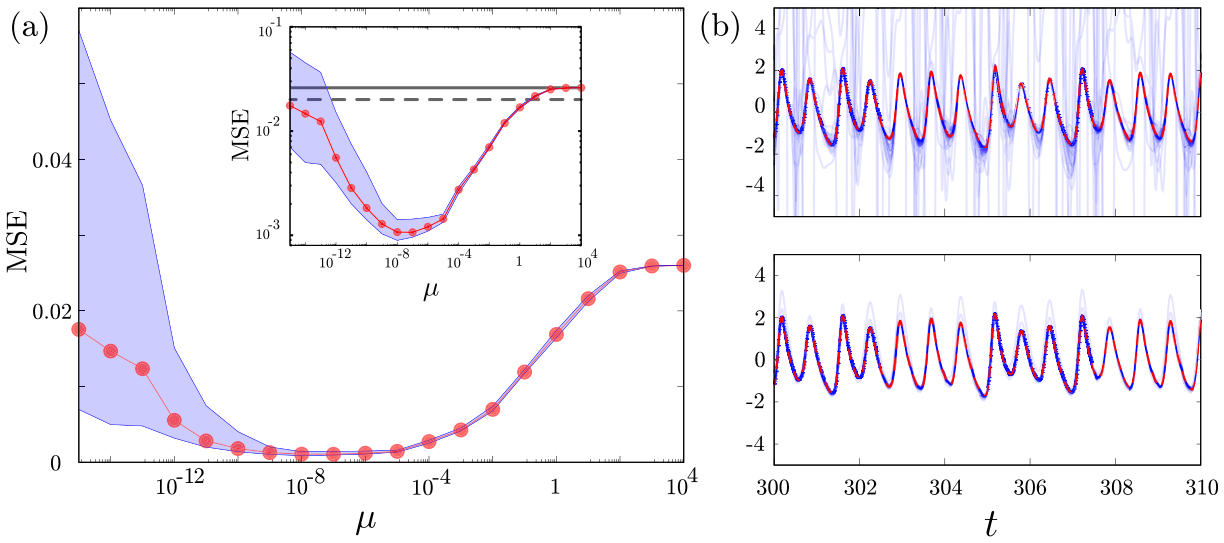

FIG. 2. (a) Generalization error (mean-square error, MSE) of the inference task for the Lorenz equations with $\rho_{\mathcal{T}} = 32$. The horizontal axis shows the transfer rate μ. The red circles represent the median MSE. The blue shaded area indicates the range of the MSEs from the first to the third quartile. Inset: the logarithmic graph of the same MSE data. The broken and solid lines represent the median MSE in the cases of $\mu = 0$ and $\mu = \infty$, respectively. (b) The blue lines are the inferred values $\hat{z}^{(j)}(t)$ $(j = 1, \cdots, 100)$ by the conventional method ($\mu = 0$, upper panel) and the proposed transfer learning ($\mu = 10^{-8}$, lower panel). The red broken lines represent the answer data $z(t)$.

The above-mentioned formula for transfer learning constitutes a one-parameter family of learning methods, which connects the conventional RC ($\mu = 0$) and the simple transfer method ($\mu \to \infty$).

III. TRANSFER LEARNING FOR LORENZ CHAOS

To verify the effectiveness of the proposed method, we use the Lorenz equations: $dx/dt = \sigma(y - x)$, $dy/dt = x(\rho - z) - y$, $dz/dt = xy - bz$. We fix the parameters σ, b to $\sigma = 10, b = 8/3$ and change the parameter ρ (Fig. 1). The task is to infer $z(t)$ from the sequence of $x(t)$ [5]. For RC, the continuous time-series $x(t)$ and $z(t)$ are sampled with a period $\tau = 0.01$, and used for the input and output signal, respectively. We assume that a sufficient amount of the training data, $\mathcal{D}_{\mathcal{S}} = \{x(t), z(t)\}_{0 \leqslant t \leqslant T_L}$, is available for $\rho_{\mathcal{S}} = 28$. We show that our method utilizes knowledge learned from $\mathcal{D}_{\mathcal{S}}$ to realize the inference in the target domains $\rho_{\mathcal{T}} = 32$ and $\rho_{\mathcal{T}} = 40$.

Figure 2(a) shows the MSE of the inference by the proposed method. The source domain is $\rho_{\mathcal{S}} = 28$ and the target domain is $\rho_{\mathcal{T}} = 32$. To evaluate the inference accuracy by the proposed method statistically, we perform the following training-testing procedure of the transfer learning:

The length of the training data $\mathcal{D}_{\mathcal{S}}$ in the source domain is $T_L = 10^4$. In the above procedure, M denotes the number of samples in the training data, $\mathcal{D}_{\mathcal{T}}{}^{(j)} = \{x^j(t), z^j(t)\}_{0 \leqslant t \leqslant T'_L}$ $(j = 1, \cdots, M)$, in the target domain.

Training-testing procedure

1: Train $\boldsymbol{w}^{\mathcal{S}}$ by Eq. (1) with $\mathcal{D}_{\mathcal{S}}$
2: Fix the transfer rate $\mu \in [0, \infty]$
3: **for** $j = 1, \cdots M$ **do**
4: Train $\delta\boldsymbol{w}^{(j)}$ by Eq. (3) with $\mathcal{D}_{\mathcal{T}}{}^{(j)}$
5: Test with $\boldsymbol{w}^{\mathcal{S}} + \delta\boldsymbol{w}^{(j)}$ and output the jth MSE
6: **end for**

We use $M = 100$ and $T'_L = 5$ $(\ll T_L)$. Each training data sample only includes less than ten cycles around the fixed points of the Lorenz attractor. Figs. 1(c) and 1(c′) present examples of the training data. The MSE is the generalization error with common test data in the target domain, which differ from the training data, with the length $T_{\text{test}} = 5 \times 10^3$. The median MSE over $M = 100$ samples of training data, indicated by the red circle, is plotted for each value of μ. The blue shaded area indicates the range of the MSE from the first to the third quartile, which characterizes the statistical dispersion of the MSE. In the inset, we show the same result for the MSE, plotted using the logarithmic values, and the dashed (solid) line represents the median MSE in the case of $\mu = 0$ ($\mu = \infty$). At each end, the proposed method is reduced to the conventional and simple transfer methods, respectively. In the case of $\mu = \infty$, we set $\delta\boldsymbol{w} = \boldsymbol{0}$.

In the case of $\rho_{\mathcal{T}} = 32$, the attractor is expected to be similar to that at $\rho_{\mathcal{S}} = 28$, and thus, transfer learning is effective. In fact, as shown in Fig. 2(a), if we conduct the transfer learning with the optimal transfer rate $\mu \simeq 10^{-8}$, then the median of the MSEs decreases drastically and it becomes smaller than, by an order of magnitude, that of the conventional method ($\mu = 0$). Note that the statistical dispersions are also reduced.

In practice, we must find the optimal parameter with a small amount of data. Even in such a situation, i.e., M is small and the length of the test data T_{test} is short, the above procedure gives an estimation of the optimal transfer rate as will be discussed in Sec. V.

The time-series of the values $\hat{z}(t)$ inferred by the conventional method and transfer learning at the optimal transfer rate are shown in the upper and lower panels in Fig. 2(b), respectively. In the figure, the solid blue lines of the inferred $\hat{z}^{(j)}(t)$ correspond to the each training data sample, $\mathcal{D}_{\mathcal{T}}{}^{(j)}$ $(j = 1, \ldots, 100)$, and the broken red line represents the answer data $z(t)$. When we use the conventional method,

the inferred time-series significantly differs from the true time-series. However, the transfer learning method enables the time-series of $z(t)$ to be inferred with a smaller error and less statistical dispersion.

Even in the case of $\rho_{\mathcal{T}} = 40$, which is far from the source domain, transfer learning can reduce the median MSE and the statistical dispersion compared with the conventional method (see Appendix C).

IV. TRANSFER LEARNING FOR FLUID TURBULENCE

We tackle a critical problem associated with turbulence: the inference of the energy dissipation rate $\epsilon(t)$. Although the energy dissipation rate plays an important role in statistical turbulence theory and turbulence modeling, the direct measurement of $\epsilon(t)$ is difficult. As an easily measurable quantity, we use the kinetic energy $K(t)$ of the turbulent flow. The task is to infer $\epsilon(t)$ from $K(t)$.

As described before, the ML-based turbulence modeling requires training data of high-Reynolds-number turbulence, such as $\{K(t), \epsilon(t)\}$, collected over a long period of time. However, such turbulence data are not available in practice. Alternatively, the calculation of turbulence at *low* Reynolds number is much easier. Since the energy cascade dynamics is insensitive to variation in the Reynolds numbers (see Ref. [14] for the details), we expect that there is a similarity of turbulence attractors over a wide range of the Reynolds numbers. Hence, the knowledge learned from turbulent data at a low Reynolds number to be useful for the same task at a high Reynolds number. This gives a reason why transfer learning is effective.

We conducted direct numerical simulations of the Navier-Stokes equations with a steady forcing term in a periodic box, using the Fourier spectral method. For the temporal integration, the fourth-order Runge–Kutta–Gill scheme was used (see Appendix D for details).

Here we use the time-series of the spatial average of the energy and the energy dissipation rate at $R_\lambda \simeq 35$ as the training data $\mathcal{D}_{\mathcal{S}} = \{K(t), \epsilon(t)\}_{0 \leqslant t \leqslant T_L}$ in the source domain, and those at $R_\lambda \simeq 120$ as the training data $\mathcal{D}_{\mathcal{T}} = \{K(t), \epsilon(t)\}_{0 \leqslant t \leqslant T'_L}$ in the target domain, where R_λ is the Taylor microscale Reynolds number and it plays the role of ρ. We emphasize that the integral-scale Reynolds number R in the target domain is approximately an order of magnitude higher than that in the source domain, since $R \propto R_\lambda^2$ [15].

Figure 3 shows the normalized time-series of the energy $K(t)$ and energy dissipation rate $\epsilon(t)$ at (a) $R_\lambda \simeq 35$ and (b) $R_\lambda \simeq 120$. The gray solid line and red broken line represent $K(t)$ and $\epsilon(t)$, respectively. The mean turnover time of the largest eddies, defined by $\langle T \rangle = \langle L/\sqrt{2K/3} \rangle$ with L being the integral length, are $\langle T \rangle \simeq 0.7$ at $R_\lambda \simeq 35$ and $\langle T \rangle \simeq 0.5$ at $R_\lambda \simeq 120$. The energy dissipation rate $\epsilon(t)$ at $R_\lambda \simeq 35$ changes in time following the energy $K(t)$ with a delay owing to the energy cascade. At $R_\lambda \simeq 120$, the time delay is still found in the relation between $K(t)$ and $\epsilon(t)$; however, the relation becomes more than merely a delay.

Training data collected for a sufficiently long time $\mathcal{D}_{\mathcal{S}}$ with $T_L = 3.0 \times 10^3$ are used to obtain $\boldsymbol{w}^{\mathcal{S}}$ at $R_\lambda \simeq 35$. We assume that the amount of available training data is highly limited for the calculation of $\delta \boldsymbol{w}$ at the target domain, $R_\lambda \simeq 120$. The

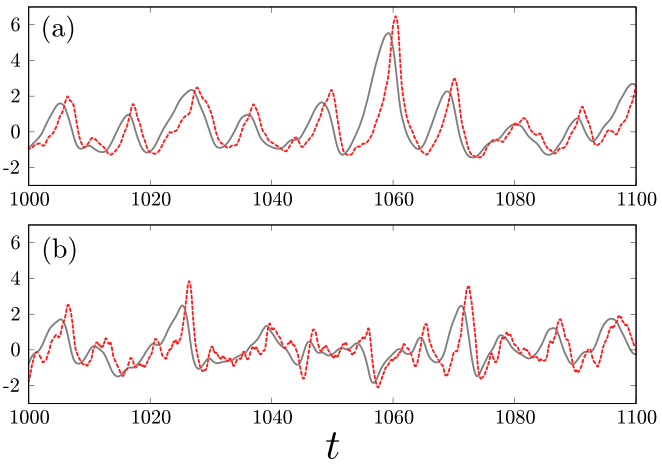

FIG. 3. Normalized time-series of the energy $K(t)$ and energy dissipation rate $\epsilon(t)$ at (a) $R_\lambda \simeq 35$ and (b) $R_\lambda \simeq 120$. The gray solid line and red broken line represent $K(t)$ and $\epsilon(t)$, respectively.

length of the each training data of $\mathcal{D}_{\mathcal{T}}{}^{(j)}$ $(j = 1, \cdots, M)$ is $T'_L = 50\,(\ll T_L)$, which includes roughly ten quasi-periodic cycles of the energy cascade events [16]. The number of samples of training data is $M = 20$. The length of the test data is $T_{\text{test}} = 2.0 \times 10^3$. For RC, the continuous time-series $K(t)$ and $\epsilon(t)$ are sampled with a period $\tau = 0.2$, and used for the input and output signal, respectively.

Figure 4(a), which uses the same symbols and lines as in Fig. 2, shows the dependency of the inference MSE on the transfer rate μ. The most accurate inference (the smallest MSE) is achieved by the proposed transfer learning method with $\mu \simeq 1$, which implies that the ESN learned from the low-Reynolds-number turbulence data requires only slight corrections by the high-Reynolds-number turbulence data. Compared with the conventional method ($\mu = 0$), transfer learning effectively reduces the inference errors and the statistical dispersion. As mentioned above, there is the similarity between the training data $\mathcal{D}_{\mathcal{S}}$ and $\mathcal{D}_{\mathcal{T}}$ for this particular task, which is explained by the energy cascade dynamics of turbulence [14].

The corresponding time-series of the inferred value $\hat{\epsilon}(t)$ in the target domain, $R_\lambda \simeq 120$, by the conventional method and transfer learning with the optimal transfer rate ($\mu = 1$) are shown in the upper and lower panels in Fig. 4(b), respectively. In each panel, the solid blue lines represent the inferred $\hat{\epsilon}^{(j)}(t)$ $(j = 1, \cdots, M)$, corresponding to each training data sample $\mathcal{D}_{\mathcal{T}}{}^{(j)}$, and the broken red line represents the answer time-series $\epsilon(t)$.

The conventional method produces a large inference error and considerable statistical dispersion. However, the transfer learning method achieves almost perfect inference of the energy dissipation rate.

V. ESTIMATION OF OPTIMAL TRANSFER RATE WITH A SMALL AMOUNT OF DATA

In the previous sections, we used a large amount of data, e.g., a large number of samples of training data and a sufficiently long test data, in order to statistically verify the performance of the transfer learning method. However, in

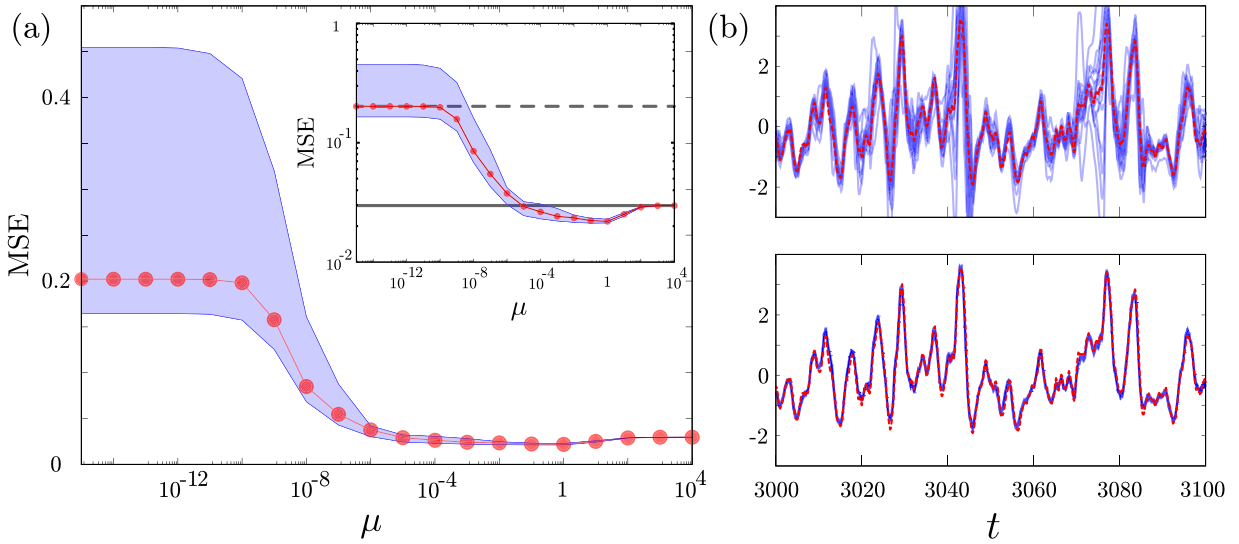

FIG. 4. (a) Generalization error (mean-square error, MSE) as a function of the transfer rate μ. Similar plots to Fig. 3 but for the inference of the energy dissipation rate of the Navier-Stokes turbulence at $R_\lambda \simeq 120$. The red circles represent the median MSE. The blue shaded area indicates the range of MSEs from the first to the third quartile. Inset: the logarithmic graph of the same MSE data. The broken and solid lines represent the median MSE in the cases of $\mu = 0$ and $\mu = \infty$, respectively. (b) The blue lines are the inferred values $\hat{\epsilon}^{(j)}(t)$ $(j = 1, \ldots, 20)$ at $R_\lambda \simeq 120$ by the conventional method ($\mu = 0$, upper panel) and the proposed transfer learning ($\mu = 1$, lower panel). The red broken lines represent the answer data $\epsilon(t)$.

practice, we need to estimate the optimal transfer rate from a small amount of data.

To investigate the optimization with a small amount of data, we conduct numerical experiments in both cases of the Lorenz equations and the Navier-Stokes equations. The target domain of the Lorenz case is $\rho_{\mathcal{T}} = 32$. Here we assume that we can use a single sample of data, i.e., $M = 1$, in the target domain with the length of $3T_L'$, and divide it into three. The first one $(0 \leqslant t < T_L')$ is used to obtain the reservoir state being synchronized with the input signal, the second one $(T_L' \leqslant t < 2T_L')$ is used for training, and the third one $(2T_L' \leqslant t < 3T_L')$ is used for testing. The other settings of the experiments including the values of T_L' for the Lorenz and Navier-Stokes equations are the same as in the previous sections.

Figures 5(a) and 5(b) show the MSE of the inference task for the Lorenz equations and for the Navier-Stokes equations, respectively. For the Lorenz equations, while the inset of Fig. 2(a) shows the optimal transfer rate is in the range $10^{-9} \leqslant \mu \leqslant 10^{-6}$, Fig. 5(a) shows it is in the range $10^{-11} \leqslant \mu \leqslant 10^{-8}$. For the Navier-Stokes equations, while the inset of Fig. 4(a) shows the optimal transfer rate is in the range $10^{-2} \leqslant \mu \leqslant 1$, Fig. 5(b) shows it is in the range $10^{-4} \leqslant \mu \leqslant 10^{-1}$. These demonstrations suggest that we can estimate the optimal transfer rate from such a small amount of data, in particular, for the turbulence case, the data for at most 15-fold of the correlation time [16] is sufficient for the optimization.

VI. CONCLUSIONS

We have developed the transfer learning method of RC and shown that the optimization of the transfer rate is essential for the inference problem of the Lorenz chaos. Furthermore, if we choose a suitable physical quantity for learning, as shown in the successful inference of the energy dissipation rate of turbulence, the amount of learning in the target domain can be drastically reduced.

In this paper, we showed in a statistically reliable manner that there exists an optimal transfer rate by using a large amount of data. However, in practice, we must find an optimal transfer rate with a small amount of data, i.e., M is small and the length of the test data T_{test} is short. As demonstrated in Sec. V, the transfer learning method gives an estimation of the optimal transfer rate even when only a small amount of data is available. To improve the accuracy of the estimation, it will be useful to employ ML-techniques such as the cross-validation or the quantification of similarity between attractors in the source and target domains.

We implemented transfer learning for parameter changes of the dynamical systems. Once we train a reservoir computer for a dynamical system with a certain parameter, the proposed method can eliminate most of the computational cost of training for the same dynamical system with different parameters. The applications of the proposed method are not restricted to the parameter change; for instance, it is possible to train a reservoir computer with numerical simulations, and then utilize it in predictions for physical experiments via the proposed transfer learning. Although the present study focused on the nonlinear dynamics, transfer learning for RC is a model-free flexible ML-method, and hence, can be applied to any other physical system. Physical RC, physical implementations of RC using physical devices such as lasers, is highly active research topic [17–19], and the transfer learning is also useful to realize the physical RC.

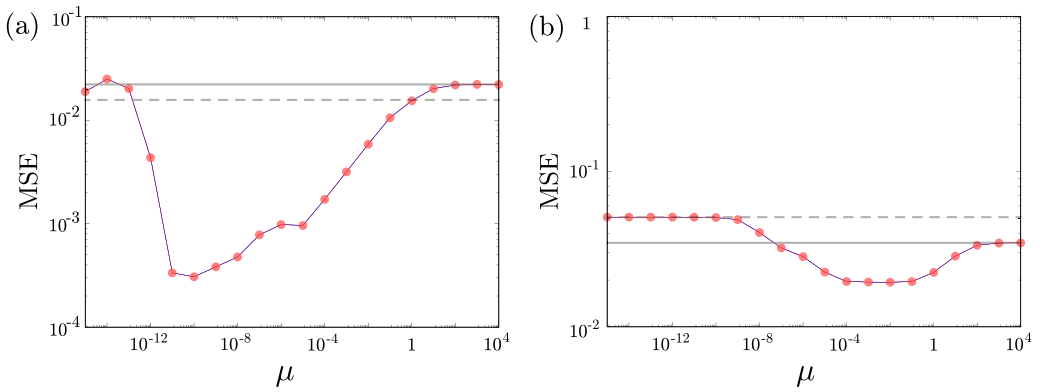

FIG. 5. MSE (red circles) of the inference task for (a) the Lorenz equations and (b) the Navier-Stokes equations, as a function of the transfer rate μ. The target domain of the Lorenz case is $\rho_{\mathcal{T}} = 32$. The MSE is estimated with a short test data $T_{\text{test}} = T'_L$ and $M = 1$. The broken and solid lines represent the MSE in the cases of $\mu = 0$ and $\mu = \infty$, respectively.

We hope that the proposed method will open up new possibilities of ML methods for nonlinear dynamics, and will be an effective tool for the long-standing problems in physics. In particular, we expect the present study to be a crucial step toward the development of the ML-based turbulence modeling that utilizes the attractor similarity associated with the universality of the small-scale statistics of turbulence.

ACKNOWLEDGMENTS

This work was partly supported by JSPS Grant-in-Aid for Early-Career Scientists No. 19K14591, JSPS Grants-in-Aid for Scientific Research No. 16H04268 and No. 20K20973, and The Nakajima Foundation. The direct numerical simulations of the Navier-Stokes equations were conducted using the supercomputer systems of the Japan Aerospace Exploration Agency (JAXA-JSS2).

APPENDIX A: REQUIRED LENGTH OF TRAINING DATA FOR ECHO STATE NETWORK

Usually, machine learning methods such as RC require a large amount of training data. In this Appendix, first, we estimate the required amount of training data explicitly in the general setting of the *linear* ESN. Then, focusing on the turbulence case studied in the main text, we discuss the required length of the training data, and describe the numerical results for the nonlinear ESN.

Training of the ESN requires the convergence of $\langle r_i r_j \rangle_t$ and $\langle r_i v \rangle_t$, where $\langle a \rangle_t := \frac{1}{t}\sum_{k=1}^{t} a(k)$. For the linear ESN, we can generally estimate the required length of training data. The linear ESN is a signal-driven dynamical system: $\boldsymbol{r}(k+1) = J\boldsymbol{r}(k) + \epsilon u(k)\mathbf{1}$, where all the components of the vector $\mathbf{1}$ are one. We obtain the following expression:

$$\begin{aligned}\boldsymbol{r}(k+1) &= J\boldsymbol{r}(k) + \epsilon u(k)\mathbf{1} \\ &= J^m \boldsymbol{r}(k-m+1) + \epsilon \sum_{\ell=0}^{m-1} u(k-\ell) J^\ell \mathbf{1} \\ &\to \epsilon \sum_{\ell=0}^{\infty} u(k-\ell) J^\ell \mathbf{1} \ (m \to \infty). \end{aligned} \tag{A1}$$

In the last step, we assume the spectral radius of the matrix $\rho(J)$ is strictly less than one, $\rho(J) < 1$, which ensures synchronization with the input signal. Therefore, we have

$$\langle r_i r_j \rangle_t = \epsilon^2 \sum_{\ell,\ell'=0}^{\infty} [J^\ell \mathbf{1}]_i [J^{\ell'} \mathbf{1}]_j C_{uu}(\ell - \ell'), \tag{A2}$$

and

$$\langle r_i v \rangle_t = \epsilon \sum_{\ell=0}^{\infty} [J^\ell \mathbf{1}]_i C_{uv}(\ell), \tag{A3}$$

where $C_{uu}(\ell - \ell')$ is the auto-correlation function, $C_{uu}(\ell - \ell') := \langle u(k-\ell)u(k-\ell') \rangle_t$, and $C_{uv}(\ell)$ is the cross-correlation function, $C_v(\ell) := \langle u(k-\ell)v(k) \rangle_t$. Thus, the convergence of $\langle r_i r_j \rangle_t$ and $\langle r_i v \rangle_t$ requires the convergence of $C_{uu}(\ell - \ell')$ and $C_{uv}(\ell)$, respectively.

Considering the inference task of the energy dissipation rate as discussed in the main text, $u = K$ and $v = \epsilon$, and sufficient training data is necessary such that the auto-correlation C_{KK} and the cross-correlation $C_{K\epsilon}$ converge. The length of such the data exceeds the turnover time of the largest eddies by several thousand times [16]. For the nonlinear ESN used in this paper, we have numerically studied the convergence of $\langle r_i r_j \rangle_t$ and $\langle r_i v \rangle_t$, and confirmed that the convergence of these values requires a long period of turbulence data as mentioned above.

APPENDIX B: PROOFS OF ASYMPTOTIC FORMULAS OF THE TRANSFER LEARNING METHODS

1. Reduction to conventional RC ($\mu = 0$)

When $\mu = 0$, Eq. (3) of the main text becomes

$$\delta \boldsymbol{w} = R'^{-1}\boldsymbol{q}' = R'^{-1}(\langle v'\boldsymbol{r}' \rangle_{T'} - C'\boldsymbol{w}^{\mathcal{S}}) = R'^{-1}\langle v'\boldsymbol{r}' \rangle_{T'} - \boldsymbol{w}^{\mathcal{S}}. \tag{B1}$$

Thus, $\boldsymbol{w}^{\mathcal{S}} + \delta\boldsymbol{w} = R'^{-1}\langle v'\boldsymbol{r}' \rangle_{T'}$, which is Eq. (1) in the main text for the conventional RC.

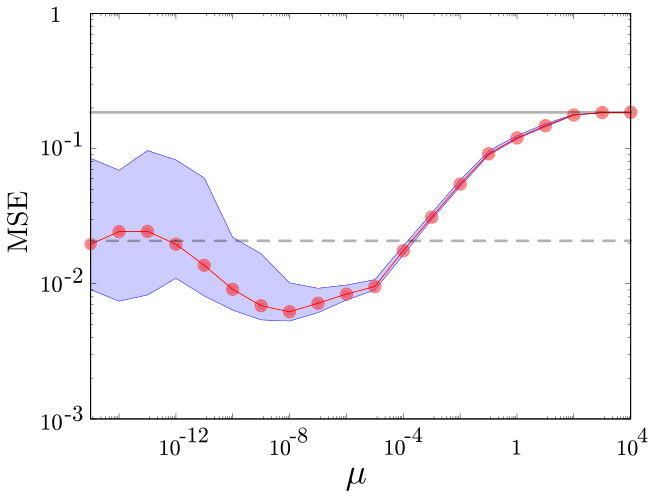

FIG. 6. MSE of the inference task for the Lorenz equations with $\rho_{\mathcal{T}} = 40$ as a function of the transfer rate μ. The red circles represent the median MSE. The blue shade indicates the range in MSEs from the first to the third quartile. The broken and solid lines represent the median MSE in the cases of $\mu = 0$ and $\mu = \infty$, respectively.

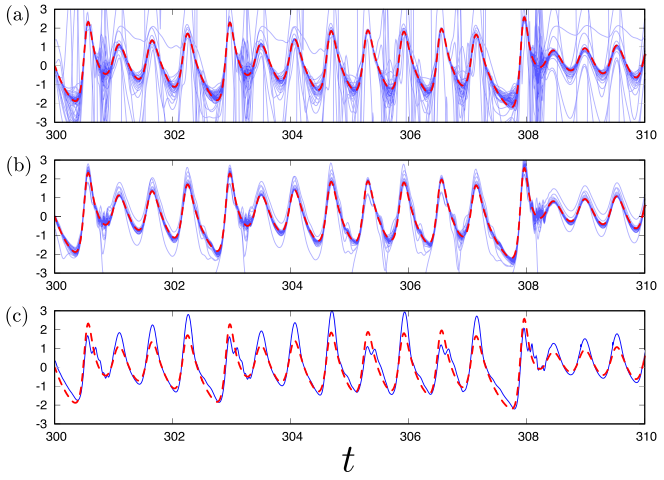

FIG. 7. The time-series of the inferred values $\hat{z}^{(j)}(t)$ $(j = 1, \ldots, M)$ at $\rho_{\mathcal{T}} = 40$ corresponding to the $M = 100$ samples of the training data. The inferred values by the conventional ($\mu = 0$), the transfer learning ($\mu = 10^{-8}$), and the simple transfer method ($\mu = \infty$) are shown in (a), (b), and (c), respectively. The red broken lines depict the answer data $z(t)$.

2. Simple transfer method ($\mu \to \infty$)

In the limit of $\mu \to \infty$, we have

$$\begin{aligned} \|\delta \boldsymbol{w}\| &= \|[R' + \mu I]^{-1}\boldsymbol{q}'\| \leqslant \|[R' + \mu I]^{-1}\| \cdot \|\boldsymbol{q}'\| \\ &= \frac{1}{\mu}\|(I - T_\mu)^{-1}\| \cdot \|\boldsymbol{q}'\| \ \ (T_\mu := -R'/\mu) \\ &\leqslant \frac{\|\boldsymbol{q}'\|}{\mu(1 - \|T_\mu\|)} \to 0 \ \ (\mu \to \infty). \end{aligned} \tag{B2}$$

In the last step, we have used the Neumann series.

APPENDIX C: TRANSFER LEARNING FOR LORENZ CHAOS FAR FROM SOURCE DOMAIN

Even in the case of $\rho_{\mathcal{T}} = 40$, which is far from the source domain, the transfer learning can reduce the median MSE and the statistical dispersion compared with the conventional method. The results are presented in Fig. 6, in which the symbols and lines have the same meaning as those in the main text.

Compared with the case of $\rho_{\mathcal{T}} = 32$, the similarity between the attractors in the source domain and the target domain are lower. In fact, the simple transfer method ($\mu = \infty$) is ineffective. Although the target domain is far from the source domain, the transfer learning with the optimal transfer rate $\mu \simeq 10^{-8}$ still reduces the median MSE and the statistical dispersions compared with the conventional method ($\mu = 0$).

The time-series of the inferred values $\hat{z}(t)$ are shown in Fig. 7. Figures 7(a), 7(b) and 7(c) correspond to the results with the conventional ($\mu = 0$), the transfer learning ($\mu = 10^{-8}$), and the simple transfer method ($\mu = \infty$), respectively. The conventional method leads to large errors and statistical dispersions. When the simple transfer method is used, inference errors around the maximal values of $z(t)$ inevitably arise. However, when the transfer learning method is used, the time-series of $z(t)$ can be inferred with a smaller error and less statistical dispersion.

APPENDIX D: DIRECT NUMERICAL SIMULATIONS OF THE NAVIER-STOKES EQUATIONS

We numerically solve the three-dimensional Navier-Stokes equations in a periodic box, using the Fourier spectral method. The aliasing errors are removed by the phase shift method. In particular, the vorticity equation,

$$\frac{\partial \boldsymbol{\omega}}{\partial t} = \nabla \times (\boldsymbol{u} \times \boldsymbol{\omega}) + \nu \nabla^2 \boldsymbol{\omega} + \nabla \times \boldsymbol{f}, \tag{D1}$$

is integrated in time with the fourth-order Runge-Kutta-Gill scheme, where $\boldsymbol{\omega} = \nabla \times \boldsymbol{u}$. The forcing term [20] is

$$\boldsymbol{f}(x, y, z) = \begin{bmatrix} -\sin(2\pi x/\mathcal{L})\cos(2\pi y/\mathcal{L}) \\ +\cos(2\pi x/\mathcal{L})\sin(2\pi y/\mathcal{L}) \\ 0 \end{bmatrix}, \tag{D2}$$

where $\mathcal{L}$ is the length of the side of the period box. The parameters of the number n^3 of the Fourier modes, the kinematic viscosity ν of the fluid, the step Δt for the temporal integration, the mean of the Taylor microscale Reynolds number $\langle R_\lambda \rangle$, and the mean turnover time of the largest eddies $\langle T \rangle$ are summarized in Table I.

TABLE I. Parameters and statistics.

Domain	n^3	ν	Δt	$\langle R_\lambda \rangle$	$\langle T \rangle$
Source	32^3	0.064	4×10^{-3}	35	0.7
Target	128^3	0.008	2×10^{-3}	120	0.5

[1] K. Duraisamy, G. Iaccarino, and H. Xiao, Turbulence modeling in the age of data, Annu. Rev. Fluid Mech. **51**, 357 (2019).
[2] M. Gamahara and Y. Hattori, Searching for turbulence models by artificial neural network, Phys. Rev. Fluids **2**, 054604 (2017).
[3] K. Fukami, Y. Nabae, K. Kawai, and K. Fukagata, Synthetic turbulent inflow generator using machine learning, Phys. Rev. Fluids **4**, 064603 (2019).
[4] K. Nakajima and I. Fischer (eds.), *Reservoir Computing: Theory, Physical Implementations, and Applications* (Springer, Singapore, 2020).
[5] Z. Lu, J. Pathak, B. Hunt, M. Girvan, R. Brockett, and E. Ott, Reservoir observers: Model-free inference of unmeasured variables in chaotic systems, Chaos **27**, 041102 (2017).
[6] R. S. Zimmermann and U. Parlitz, Observing spatio-temporal dynamics of excitable media using reservoir computing, Chaos **28**, 043118 (2018).
[7] K. Nakai and Y. Saiki, Machine-learning inference of fluid variables from data using reservoir computing, Phys. Rev. E **98**, 023111 (2018).
[8] A. Cunillera, M. C. Soriano, and I. Fischer, Cross-predicting the dynamics of an optically injected single-mode semiconductor laser using reservoir computing, Chaos **29**, 113113 (2019).
[9] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach, Phys. Rev. Lett. **120**, 024102 (2018).
[10] M. Inubushi and K. Yoshimura, Reservoir computing beyond memory-nonlinearity trade-off, Sci. Rep. **7**, 10199 (2017).
[11] K. Weiss, T. M. Khoshgoftaar, and D. Wang, A survey of transfer learning, J. Big Data **3**, 9 (2016).
[12] M. Inubushi and S. Goto, Transferring reservoir computing: Formulation and application to fluid physics, in *International Conference on Artificial Neural Networks* (Springer, Berlin, 2019), pp. 193–199.
[13] H. Jaeger, The "echo state" approach to analyzing and training recurrent neural networks—with an erratum note, Technical Report, Vol. 148 (German National Research Center for Information Technology GMD, Bonn, Germany, 2001), p. 13.
[14] S. Goto and M. Inubushi (unpublished).
[15] U. Frisch, *Turbulence: The Legacy of AN Kolmogorov* (Cambridge University Press, Cambridge, UK, 1995).
[16] S. Goto and J. C. Vassilicos, Local equilibrium hypothesis and Taylor's dissipation law, Fluid Dyn. Res. **48**, 021402 (2016).
[17] S. Sunada and A. Uchida, Photonic reservoir computing based on nonlinear wave dynamics at microscale, Sci. Rep. **9**, 19078 (2019).
[18] I. Estébanez, I. Fischer, and M. C. Soriano, Constructive Role of Noise for High-Quality Replication of Chaotic Attractor Dynamics Using a Hardware-Based Reservoir Computer, Phys. Rev. Appl. **12**, 034058 (2019).
[19] G. Tanaka, T. Yamane, J. B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano, and A. Hirose, Recent advances in physical reservoir computing: A review, Neural Netw. **115**, 100 (2019).
[20] S. Goto, Y. Saito, and G. Kawahara, Hierarchy of antiparallel vortex tubes in spatially periodic turbulence at high Reynolds numbers, Phys. Rev. Fluids **2**, 064603 (2017).