

# 博士論文

論文題目 Machine-learning Construction of a Model and  
Refined Regularity Criterion on Fluid  
Equations  
(流体方程式の機械学習によるモデリングと  
解の正則性の判定条件)

氏名 中井 拳吾

# Preface

It is known that most fluid flow is governed by the Navier–Stokes equations. The equation has lots of difficulties; nonlinearity, nonlocal interactions.

In this paper, we consider two topics. One of them is machine-learning construction of a model for fluid dynamics, the other one is refined regularity criterion for generalized Navier–Stokes equations.

We consider the three-dimensional incompressible Navier–Stokes equations:

$$\begin{aligned} \partial_t v + (v \cdot \nabla)v &= -\nabla\pi + \nu\Delta v && \text{in } \mathbb{R}^3 \times (0, \infty) \\ \operatorname{div} v &= 0 && \text{in } \mathbb{R}^3 \times (0, \infty) \\ v(x, 0) &= v_0(x) && \text{in } \mathbb{R}^3, \end{aligned}$$

where  $v = v(x, t) = (v_1(x, t), v_2(x, t), v_3(x, t))$  is the velocity of the fluid flows,  $\pi = \pi(x, t)$  is the pressure,  $\nu > 0$  is viscosity parameter, and  $v_0(x)$  is a given initial velocity field satisfying  $\operatorname{div} v_0 = 0$ . J. Leary (1934) and E. Hopf (1951) proved the existence of weak solutions of the equations. The existence of global regular solution and the uniqueness with large initial field is one of the Millennium Prize Problems which were stated by the Clay Mathematics Institute.

It is important describing the behavior of macroscopic variable such as the energy which is defined by  $\|v\|_{L^2}$ , when we consider the problem. It is hard to derive an equation of macroscopic variables analytically from the Navier–Stokes equations of microscopic variables. For example, in order to describe a dynamics of  $n$ -th moment variables, we have to use  $n + 1$ -th moment variables, which is called closure problem. Until now, many researchers construct fluid models of macroscopic variables, however, most of them are obtained with the nonlinear term of the Navier–Stokes equations replaced by some posteriori term.

Here, our aim of investigation is construction of a model of macroscopic variables for fluid flow by using Machine-learning. In this construction, we consider the Energy functions and the Reynolds number as macroscopic variables.

In this paper, the “learning” of machine-learning corresponds to the finding of the relation “ $F_1$ ” ( $F_1 : \mathbb{R}^M \rightarrow \mathbb{R}^M$ ) between  $\mathbf{u} \in \mathbb{R}^M$  and  $\mathbf{s}(t + \Delta t) \in \mathbb{R}^M$ , which satisfies

$$F_1(\mathbf{u}(t)) \approx \mathbf{s}(t + \Delta t) \quad \text{for any } t \in [0, T], \quad (0.0.1)$$

where  $\mathbf{u}$  is fluid data at the time  $t$  and  $\mathbf{s}(t + \Delta t)$  is data at the time  $t + \Delta t$ , which corresponds to the next step of time  $t$ . When we find “good” relation  $F_1$ , for the new data  $\mathbf{u}(T + 1)$ ,

$$\hat{\mathbf{s}}(T + 1 + \Delta t) = F_1(\mathbf{u}(T + 1)),$$

is similar to  $\mathbf{s}(T + 1 + \Delta t)$  which is the known data. This means that we succeeded in predicting a  $\mathbf{s}(T + 1 + \Delta t)$ .

In this paper, we deal with a time-series data and so high dimensional dynamical system. So, we use the following form instead of eq. (0.0.1);

$$F_2(F_3(\mathbf{u}(t), \mathbf{r}(t))) \approx \mathbf{s}(t + \Delta t) \quad \text{for any } t \in [0, T], \quad (0.0.2)$$

where  $\mathbf{r}(t) = F_3(\mathbf{u}(t - \Delta t), \mathbf{r}(t - \Delta t)) \in \mathbb{R}^N$  and  $N > M$ .

The vector  $\mathbf{r}(t^* + \Delta t) (= F_3(\mathbf{u}(t^*), \mathbf{r}(t^*)))$  is influenced by the time-series data  $\{\mathbf{u}(t)\}_{t=0}^{t^*}$ . This means that the vector  $\mathbf{r}$  include the information of past states of  $\mathbf{u}$ . Also, due to setting  $N > M$ , we can decompose the input data  $\mathbf{u}$ . In this paper,  $\mathbf{u}$  is macroscopic variables.

In our learning, we do not adjust a relation “ $F_3$ ” to save a great amount of computational costs. Instead, we set a sufficiently large number as  $N(\gg M)$ , which enables us to deal with a complex deterministic behavior. This way of learning is called reservoir computing. What is done is determination of output layer  $F_2$  between reservoir states  $\mathbf{r}$  and output data  $\mathbf{s}$  in eq. (0.0.2).

Recently it is reported that the reservoir computing is effective in the inference of time-series and some characteristics using the Lorenz system, Rossler system and Kuramoto-Sivashinsky system (J. Pathak, Z. Lu, B. Hunt, M. Girvan, and E. Ott, *Chaos* 27, 121102 (2017)).

In Chapter 1, we infer macroscopic behaviors of a three-dimensional fluid flow with chaotic behaviors using reservoir computing. This chapter is based on the following paper:

- K. Nakai and Y. Saiki, Machine-learning inference of fluid variables from data using reservoir computing, *Physical Review E* 98, 023111:1-6, 2018.

In our procedure of the inference, we assume no prior knowledge of a physical process of a fluid flow except that its behavior is complex but deterministic. We present an inference of the complex behavior, which requires only past time-series data as training data.

We show that the reservoir dynamics constructed from only past data of energy functions can infer the future behavior of energy functions and reproduce the energy spectrum. The energy function  $E_0(k, t)$  for wavenumber  $k \in \mathbb{N}$  and at time  $t$  is defined by

$$E_0(k, t) := \frac{1}{2} \sum_{\kappa \in D_k} \sum_{\zeta=1}^3 |\mathcal{F}_{[v_\zeta]}(\kappa, t)|^2,$$

where  $D_k := \{\kappa \in \mathbb{Z}^3 | k - 0.5 \leq |\kappa| < k + 0.5\}$  and  $\mathcal{F}_{[f]}$  is the Fourier transform of  $f$ .  $F_2$  is determined by setting

$$\begin{aligned} \mathbf{u}(t) &= (\tilde{E}(1, t), \tilde{E}(2, t), \dots, \tilde{E}(9, t))^t, \\ \mathbf{s}(t) &= (\tilde{E}(1, t), \tilde{E}(2, t), \dots, \tilde{E}(9, t))^t, \end{aligned}$$

where  $\tilde{E}(\cdot, \cdot)$  is normalized value of  $E(\cdot, \cdot)$ . We found that an inference of energy functions is successful for some time after finishing training 9-dimensional time-series data of energy functions.

Moreover, it is also shown that we can infer a time-series data from only one measurement by using the delay coordinates. These imply that the obtained two reservoir systems constructed without the knowledge of microscopic data are equivalent to the dynamical systems describing macroscopic behavior of energy functions.

In Chapter 2, We construct a data-driven dynamical system model for the Taylor microscale Reynolds number of a high-dimensionally chaotic fluid flow by training its scalar time-series data. This chapter is based on the following paper:

- K. Nakai and Y. Saiki, Machine-learning construction of a model for a macroscopic fluid variable using the delay-coordinate of a scalar observable, *Discrete and Continuous Dynamical Systems Series S* (Accepted).

The Reynolds number, which represents the degree of complexity of a fluid flow at time  $t$ , is defined by

$$\tilde{R}_\lambda(t) := \sqrt{\frac{20E(t)^2}{3\nu\epsilon(t)}},$$

where,

$$\epsilon(t) = 2\nu \sum_{\kappa \in D} \sum_{\zeta=1}^3 |\kappa|^2 (\mathcal{F}_{[v_\zeta]}(\kappa, t))^2,$$

is the average rate of energy dissipation per unit mass,  $E(t) = \sum_k E_0(k, t)$  is the total energy at time  $t$ .

Here, in Chapter 1, we see that we can infer a time-series data from only one measurement by using the delay coordinates. So,  $F_2$  is determined by setting

$$\begin{aligned} \mathbf{u}(t) &= (\tilde{R}_\lambda(t), \tilde{R}_\lambda(t - \Delta\tau), \dots, \tilde{R}_\lambda(t - (M - 1)\Delta\tau))^T, \\ \mathbf{s}(t) &= (\tilde{R}_\lambda(t), \tilde{R}_\lambda(t - \Delta\tau), \dots, \tilde{R}_\lambda(t - (M - 1)\Delta\tau))^T, \end{aligned}$$

where  $\tilde{R}_\lambda(\cdot)$  is normalized value of  $E(k, t)$ . We found that an inference of the Reynolds number is successful for some time.

Moreover, we investigate the appropriate choice of the delay-coordinate, especially the delay-time  $\Delta\tau$  and the dimension  $M$ , which enables us to construct a model having a relatively high-dimensional attractor with low computational costs. The appropriate choice will be discussed in Sec. 2.5.

In Chapter 3, we consider a regularity criterion for generalized Navier–Stokes equations. This chapter is based on the following paper:

- K. Nakai, Direction of Vorticity and a Refined Regularity Criterion for the Navier–Stokes Equations with Fractional Laplacian, *Journal of Mathematical Fluid Mechanics* 21, 21, 2019.

We define the vorticity  $\omega$  by using the differential operator  $\text{rot}$ ;

$$\omega = \text{rot}v.$$

For the solution  $v \in E_s(T)$  of the Navier–Stokes equations, if the vorticity  $\omega$  belongs to  $L^1(0, T; L^\infty(\mathbb{R}^3))$ , the velocity  $v$  can be continued to the strong solution in the class  $E_s(T^*)$  for some  $T < T^*$ , where the class  $E_s(T^*)$  is defined as following by using Sobolev space  $H^s$ ;

$$E_s(T) := C([0, T]; H^s) \cap C^1([0, T]; H^{s-1}) \quad (s \geq 3).$$

J. T. Beale, T. Kato, A. Majda (1984) proved the above theorem for the Euler equations. However we can also prove that for the Navier–Stokes equations.

P. Constantin, C. Fefferman [9] considered the continuity of direction vector of vorticity

$$\frac{\sqrt{1 - (\xi(x, t) \cdot \xi(x + h, t))^2}}{|h|^\beta} \quad (=:\eta_\beta(x, h, t)). \quad (0.0.3)$$

They proved that the Lipschitz continuity  $\eta_\beta$  with  $\beta = 1$  induces the regularity of velocity to the Navier–Stokes equations. On the other hand, H. Beirão da Veiga, L. Berselli [3, 2] proved that the  $\beta$ -Hölder continuity  $\eta_\beta$  with  $\beta = 1/2$  induces the regularity of velocity.

With these previous results in mind, we consider the following generalized Navier–Stokes equations:

$$\partial_t v + (v \cdot \nabla)v = -\nabla\pi - \nu(-\Delta)^{\alpha/2}v \quad \text{in } \mathbb{R}^3 \times (0, \infty) \quad (0.0.4)$$

$$\text{div } v = 0 \quad \text{in } \mathbb{R}^3 \times (0, \infty) \quad (0.0.5)$$

$$v(x, 0) = v_0(x) \quad \text{in } \mathbb{R}^3. \quad (0.0.6)$$

Here, a general fractional Laplacian  $(-\Delta)^{\alpha/2}$  ( $\alpha > 0$ ) is defined by

$$\mathcal{F} [(-\Delta)^{\alpha/2}f](\xi) = |\xi|^\alpha \mathcal{F}_{[f]}(\xi),$$

where  $\mathcal{F}_{[f]}$  is the Fourier transform of  $f$ . We denote the equations (0.0.4)–(0.0.6) by  $(\text{NS})_\alpha$ . When  $\alpha = 2$ , the  $(\text{NS})_\alpha$  reduce to the usual Navier–Stokes equations. In this paper we are concerned with the case  $0 < \alpha \leq 2$ .

The  $(\text{NS})_\alpha$  were first considered by J.L. Lions [28]. It is known in [48] that when  $\alpha > 0$ ,  $(\text{NS})_\alpha$  with  $v_0 \in L^2$  possess a global weak solution. Furthermore, if  $\alpha \geq 5/2$ ,  $(\text{NS})_\alpha$  have a unique global smooth solution [28].

In this thesis, we prove a refined regularity criterion for  $(\text{NS})_\alpha$  (Teorem 3.1.4).

Let  $\beta \in (0, 1]$ ,  $a, b, q \in (1, \infty]$ ,  $r \in (1, 3/\beta)$  satisfy

$$\frac{1}{b} + \frac{1}{r} < \frac{\alpha + \beta}{3}, \quad \frac{\alpha}{a} + \frac{\alpha}{q} + \frac{3}{b} + \frac{3}{r} \leq \alpha + \beta.$$

Suppose that

- $\omega \in L^q(0, T; L^r)$
- there exist  $g \in L^a(0, T; L^b)$  such that  $\eta_\beta \leq g(x, t)$  in regions of high vorticity

Then  $v$  can be continued to the strong solution in the class  $E_s(T^*)$  for some  $T < T^*$ .

Roughly speaking, this statement means that the assumption of regularity of the direction vector of vorticity in regions of high vorticity induces the regularity of velocity to  $(\text{NS})_\alpha$  as in the case of the usual Navier–Stokes equations.

# Acknowledgement

I would like to thank my supervisor, Professor Tsuyoshi Yoneda, for his helpful advice and encouragement. I would also like to thank Professor Yoshitaka Saiki at Hitotsubashi University, for his helpful advice, discussions and collaboration on the contents of Chapter 1 and 2. I also thank to the secondary supervisor in the Program for Leading Graduate Schools, Professor Norikazu Saito. On the contents of Chapter 3, I would like to thank to Professor Yong Zhou for telling me information of the Navier–Stokes equations with fractional powers of the Laplacian.

This thesis was supported by the Leading Graduate Course for Frontiers of Mathematical Sciences and Physics (FMSP) at the University of Tokyo and grant-in-aid for JSPS fellow 19J12482. Part of the computation was supported by the Collaborative Research Program for Young · Women Scientists of ACCMS and IIMC, Kyoto University and by “High Performance Computing Infrastructure” in Japan (Project ID: jh190070).

# Contents

<b>Introduction</b>	<b>1</b>
<b>Acknowledgement</b>	<b>4</b>
<b>1 Machine-learning inference of fluid variables</b>	<b>6</b>
1.1 Introduction . . . . .	6
1.2 Reservoir computing . . . . .	7
1.3 Fluid flow . . . . .	8
1.4 Partial-inference of microscopic variables: Fourier variables of velocity. . . . .	9
1.5 Full-inference of macroscopic variables: Energy function and Energy spectrum . . . . .	10
1.6 Full-inference of Macroscopic variable from only one measurement using delay coordinates	12
1.7 Discussion and remarks . . . . .	13
1.8 Acknowledgements . . . . .	14
<b>2 Machine-learning construction using the delay-coordinate of a scalar observable</b>	<b>15</b>
2.1 Introduction . . . . .	15
2.2 Reservoir computing . . . . .	16
2.3 Generation of a fluid flow data . . . . .	17
2.4 Construction of a model for a macroscopic variable: Reynolds Number . . . . .	19
2.4.1 Construction of variables based on delay-coordinate . . . . .	19
2.4.2 Evaluation of the model . . . . .	19
2.5 Choice of delay-coordinate. . . . .	25
2.6 Summary and discussion . . . . .	26
2.7 Acknowledgements . . . . .	27
<b>3 Refined Regularity Criterion</b>	<b>28</b>
3.1 Introduction . . . . .	28
3.2 Deformation of the convection term . . . . .	31
3.3 Proof of Theorem 3.1.4 . . . . .	31
<b>Bibliography</b>	<b>35</b>

# Chapter 1

## Machine-learning inference of fluid variables from data using reservoir computing

**Abstract:** We infer both microscopic and macroscopic behaviors of a three-dimensional chaotic fluid flow using reservoir computing. In our procedure of the inference, we assume no prior knowledge of a physical process of a fluid flow except that its behavior is complex but deterministic. We present two ways of inference of the complex behavior; the first called partial-inference requires continued knowledge of partial time-series data during the inference as well as past time-series data, while the second called full-inference requires only past time-series data as training data. For the first case, we are able to infer long-time motion of microscopic fluid variables. For the second case, we show that the reservoir dynamics constructed from only past data of energy functions can infer the future behavior of energy functions and reproduce the energy spectrum. It is also shown that we can infer a time-series data from only one measurement by using the delay coordinates. These implies that the obtained two reservoir systems constructed without the knowledge of microscopic data are equivalent to the dynamical systems describing macroscopic behavior of energy functions.

### 1.1 Introduction

Machine-learning has progressed significantly over the last decade in various areas of physical sciences [39, 14, 33] after some theoretical works in the area of neural networks (See [17, 10] for examples.)

In fluid dynamics area Ling *et al.* [27] presents a method of using deep neural networks to learn a model for the Reynolds stress anisotropy tensor from high-fidelity simulation data (see also [26]). Gamahara and Hattori [15] uses an artificial neural network to find a new subgrid model of the subgrid-scale stress in large-eddy simulation. By using “Long Short-Term Memory (LSTM)” [16], Wan *et al.* [47] studies a data-assisted reduced-order modeling of extreme events in various dynamics including the Kolomogorov flow of the two-dimensional incompressible Navier–Stokes equation. See also Vlachas *et al.* [46] for the result on the barotropic climate model.

It is recently reported that reservoir computing, brain-inspired machine-learning framework that employs a data-driven dynamical system, is effective in the inference of a future such as time-series, frequency spectra and the Lyapunov spectra [45, 19, 30, 36, 18, 35, 1]. Pathak *et al.* [36] exemplifies using the Lorenz system and the Kuramoto-Sivashinsky system that the model obtained by reservoir computing can generate an arbitrarily long time-series whose Lyapunov exponents approximate those of the input signal.

A reservoir is a recurrent neural network whose internal parameters are not adjusted to fit the data in the training process. What is done is to train the reservoir by feeding it an input time-series and fitting a linear function of the reservoir state variables to a desired output time-series. Due to this approach of reservoir computing we can save a great amount of computational costs, which enables us to deal with a complex deterministic behavior. The framework was proposed as Echo-State Networks [22, 23] and Liquid-State Machines [32].

It is known that an inference of a fluid flow is difficult but important in both physical and industrial aspects. In this chapter, we infer variables of a chaotic fluid flow by applying the method of reservoir computing without a prior knowledge of physical process.

## 1. Machine-learning inference of fluid variables

After introducing the method of reservoir computing in Sec. 1.2 and a fluid flow in Sec. 1.3, we explain how to apply the method to the inference of fluid variables, and show that inferences of both microscopic and macroscopic behaviors are successful in Sec. 1.4 and Sec. 1.5, respectively. In Sec. 1.6, we exemplify that a time-series inference of high-dimensional dynamics is possible by using delay coordinates, even when the number of measurements is smaller than the Lyapunov dimension of the attractor. Discussions and remarks are given in Sec. 1.7.

### 1.2 Reservoir computing

Reservoir computing is recently used in the inference of complex dynamics [30, 36, 35, 18, 29]. The reservoir computing focuses on the determination of a translation matrix from reservoir state variables to variables to be inferred (see eq. (1.2.4)). Here we review the outline of the method [23, 30]. We consider a dynamical system

$$\frac{d\phi}{dt} = \mathbf{f}(\phi),$$

together with a pair of  $\phi$ -dependent, vector valued variables

$$\mathbf{u} = \mathbf{h}_1(\phi) \in \mathbb{R}^M \quad \text{and} \quad \mathbf{s} = \mathbf{h}_2(\phi) \in \mathbb{R}^P.$$

We seek a method for using the continued knowledge of  $\mathbf{u}$  to determine an estimate of  $\mathbf{s}$  as a function of time when direct measurement of  $\mathbf{s}$  is not available, which we call the **partial-inference**. We also consider the **full-inference** for which we have a knowledge  $\mathbf{u}$  only for  $t \leq T$ . Concerning the algorithm, this is just a variant of the partial-inference [36, 35], and will be explained later.

The dynamics of the reservoir state vector

$$\mathbf{r} \in \mathbb{R}^N \quad (N \gg M),$$

is defined by

$$\mathbf{r}(t + \Delta t) = (1 - \alpha)\mathbf{r}(t) + \alpha \tanh(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{\text{in}}\mathbf{u}(t)), \quad (1.2.1)$$

where  $\Delta t$  is a relatively short time step. The matrix  $\mathbf{A}$  is a weighted adjacency matrix of the reservoir layer, and the  $M$ -dimensional input  $\mathbf{u}(t)$  is fed in to the  $N$  reservoir nodes via a linear input weight matrix denoted by  $\mathbf{W}_{\text{in}}$ . The parameter  $\alpha$  ( $0 < \alpha \leq 1$ ) in eq. (1.2.1) adjusts the nonlinearity of the dynamics of  $\mathbf{r}$ , and is chosen depending upon the complexity of the dynamics of measurements and the time step  $\Delta t$ .

Each row of  $\mathbf{W}_{\text{in}}$  has one nonzero element, chosen from a uniform distribution on  $[-\sigma, \sigma]$ . The matrix  $\mathbf{A}$  is chosen from a sparse random matrix in which the fraction of nonzero matrix elements is  $(D_1 + D_2)/N$ , so that the average degree of a reservoir node is  $D_1 + D_2$ . The  $D_1$  non-zero components are chosen from a uniform distribution on  $[-1, 1]$ , and  $D_2$  from that on  $[-\gamma, \gamma]$  for  $\gamma (\ll 1)$ , where  $D_2$  non-zero components are introduced to reflect weak couplings among components of  $\mathbf{r}$ . Then we uniformly rescale all the elements of  $\mathbf{A}$  so that the largest value of the magnitudes of its eigenvalues becomes  $\rho$ .

The output, which is a  $P$ -dimensional vector, is taken to be a linear function of the reservoir state  $\mathbf{r}$ :

$$\hat{\mathbf{s}}(t) = \mathbf{W}_{\text{out}}\mathbf{r}(t) + \mathbf{c}. \quad (1.2.2)$$

The reservoir state  $\mathbf{r}$  evolves following eq. (1.2.1) with input  $\mathbf{u}(t)$ , starting from random initial state  $\mathbf{r}(-\tau)$  whose elements are chosen from  $(0, 1]$  in order not to diverge, where  $\tau/\Delta t (\gg 1)$  is the transient time. We obtain  $L = T/\Delta t$  steps of reservoir states  $\{\mathbf{r}(l\Delta t)\}_{l=1}^L$  by eq. (1.2.1). Moreover, we record the actual measurements of the state variables  $\{\mathbf{s}(l\Delta t)\}_{l=1}^L$ .

We train the network by determining  $\mathbf{W}_{\text{out}}$  and  $\mathbf{c}$  so that the reservoir output approximates the measurement for  $0 < t \leq T$  (training phase), which is the main part of this computation. We do this by minimizing the following quadratic form with respect to  $\mathbf{W}_{\text{out}}$  and  $\mathbf{c}$ :

$$\sum_{l=1}^L \|(\mathbf{W}_{\text{out}}\mathbf{r}(l\Delta t) + \mathbf{c}) - \mathbf{s}(l\Delta t)\|^2 + \beta[\text{Tr}(\mathbf{W}_{\text{out}}\mathbf{W}_{\text{out}}^T)], \quad (1.2.3)$$

## 1. Machine-learning inference of fluid variables

where  $\|\mathbf{q}\|^2 = \mathbf{q}^T \mathbf{q}$  for a vector  $\mathbf{q}$ , and the second term is a regularization term introduced to avoid overfitting  $\mathbf{W}_{\text{out}}$  for  $\beta \geq 0$ . When the training is successful,  $\hat{\mathbf{s}}(t)$  should approximate the desired unmeasured quantity  $\mathbf{s}(t)$  for  $t > T$  (inference phase). Following eq. (1.2.2), we obtain

$$\hat{\mathbf{s}}(t) = \mathbf{W}_{\text{out}}^* \mathbf{r}(t) + \mathbf{c}^*, \quad (1.2.4)$$

where  $\mathbf{W}_{\text{out}}^*$  and  $\mathbf{c}^*$  denote the solutions for the minimizers of the quadratic form (1.2.3) (see [31] P.140 for details):

$$\begin{aligned} \mathbf{W}_{\text{out}}^* &= \delta \mathbf{S} \delta \mathbf{R}^T (\delta \mathbf{R} \delta \mathbf{R}^T + \beta \mathbf{I})^{-1}, \\ \mathbf{c}^* &= -[\mathbf{W}_{\text{out}}^* \bar{\mathbf{r}} - \bar{\mathbf{s}}], \end{aligned}$$

where  $\bar{\mathbf{r}} = \sum_{l=1}^L \mathbf{r}(l\Delta t)/L$ ,  $\bar{\mathbf{s}} = \sum_{l=1}^L \mathbf{s}(l\Delta t)/L$ , and  $\mathbf{I}$  is the  $N \times N$  identity matrix,  $\delta \mathbf{R}$  (respectively,  $\delta \mathbf{S}$ ) is the matrix whose  $l$ -th column is  $\mathbf{r}(l\Delta t) - \bar{\mathbf{r}}$  (respectively,  $\mathbf{s}(l\Delta t) - \bar{\mathbf{s}}$ ).

In order to consider the effect of all the variables equally, we take the normalized value  $\tilde{X}(t)$  for each

parameter		(a)	(b)	(c)
$\tau$	transient time	1000	2500	2350
$T$	training time	10000	20000	20000
$M$	dimension of measurements	270	9	36
$P$	dimension of inferred variables	2	9	36
$N$	number of reservoir nodes	6400	3200	3200
$D_1$	parameter of determining elements of $\mathbf{A}$	60	320	120
$D_2$	parameter of determining elements of $\mathbf{A}$	60	0	0
$\gamma$	scale of input weights in $\mathbf{A}$	0.1	0	0
$\rho$	maximal eigenvalue of $\mathbf{A}$	1.0	0.5	0.5
$\sigma$	scale of input weights in $\mathbf{W}_{in}$	0.4	0.3	0.5
$\alpha$	nonlinearity degree of reservoir dynamics	0.7	0.3	0.4
$\Delta t$	time step for reservoir dynamics	0.1	0.25	0.5
$\beta$	regularization parameter	0	0.01	0.1

Table 1.1: **Sets of parameters for our reservoir computing.** The set (a) is used for the partial-inference of microscopic Fourier variables, whereas the set (b) is for the full-inference of macroscopic variables of energy functions and energy spectrum, and the set (c) is for the full-inference from only one measurement.

variable  $X(t)$ , which will be used throughout the whole procedure of our reservoir computing:

$$\tilde{X}(t) = [X(t) - X_1]/X_2,$$

where  $X_1$  is the mean value and  $X_2$  is the variance. When we reconstruct  $X(t)$  in the inference phase from  $\tilde{X}(t)$ , we employ  $X_1$  and  $X_2$  obtained in the training phase. Due to the normalization we can avoid adjustments of  $\sigma$ .

## 1.3 Fluid flow

In order to generate measurements of the reservoir computing, we employ the direct numerical simulation of the incompressible three-dimensional Navier–Stokes equation under periodic boundary conditions:

$$\begin{cases} \partial_t v - \nu \Delta v + (v \cdot \nabla) v + \nabla \pi = f, & \nabla \cdot v = 0, & \mathbb{T}^3 \times (0, \infty), \\ v|_{t=0} = v_0 & \text{with } \nabla \cdot v_0 = 0, & \mathbb{T}^3, \end{cases}$$

where  $\mathbb{T} = [0, 2\pi)$ ,  $\nu > 0$  is viscosity parameter,  $\pi(x, t)$  is pressure, and  $v(x, t) = (v_1(x, t), v_2(x, t), v_3(x, t))$  is velocity. We use the Fourier spectral method [21] with  $N_0 (= 9)$  modes in each direction, meaning that the system is approximated by  $2(2N_0 + 1)^3 (= 13718)$ -dimensional ordinary differential equations (ODEs).

## 1. Machine-learning inference of fluid variables

The ODEs are integrated by the 4th-order Runge–Kutta method, and the forcing is input into the low-frequency variables at each time step so as to preserve the energy of the low-frequency part. That is, both the real and the imaginary parts of the Fourier coefficient of the vorticity  $\omega$  ( $= \text{rot } v$ ),

$$\mathcal{F}_{[\omega_\zeta]}(\kappa, t) := \frac{1}{(2\pi)^3} \int_{\mathbb{T}^3} \omega_\zeta(x, t) e^{-i(\kappa \cdot x)} dx,$$

are kept constant for  $\zeta = 1, 2$ ,  $\kappa = (1, 0, 0), (0, 1, 0)$ . We use an initial condition, which has energy only in the low-frequency variables. See [21] for the details.

### 1.4 Partial-inference of microscopic variables: Fourier variables of velocity.

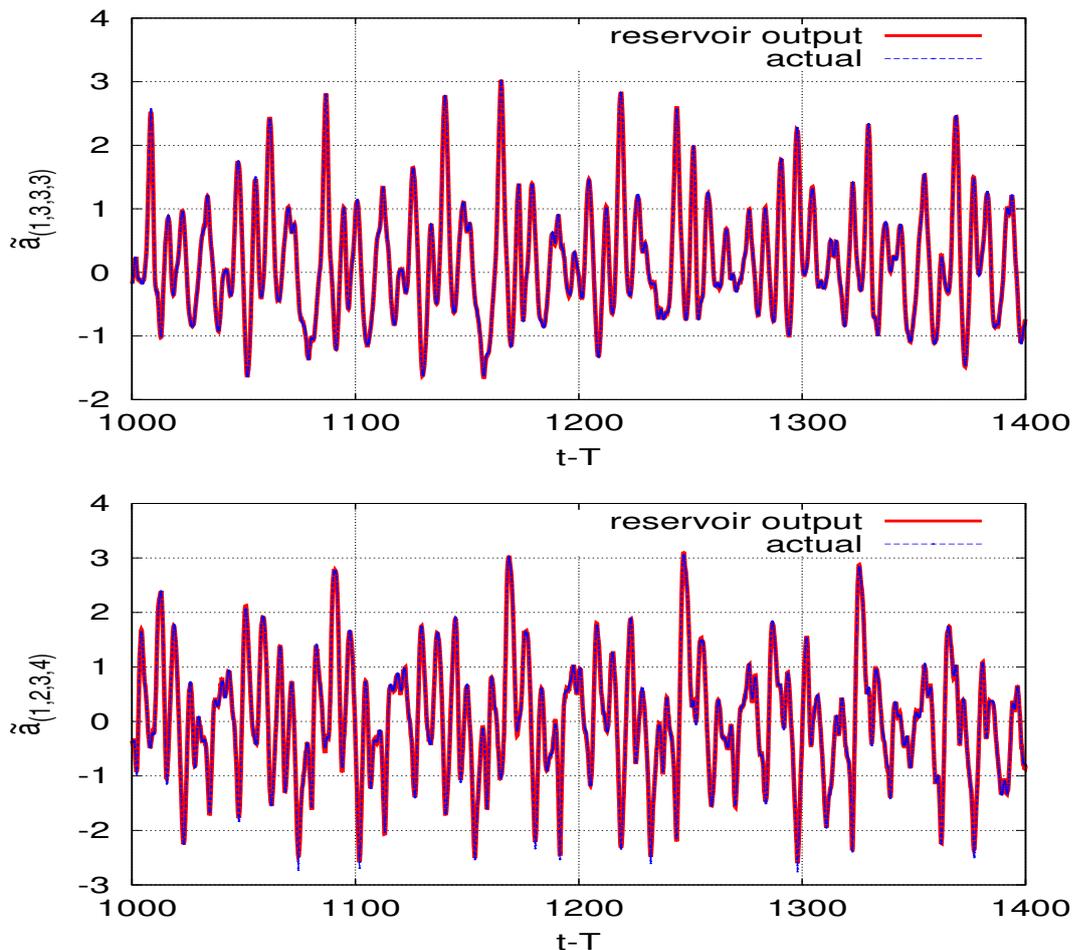


Figure 1.1: **Partial-inference of time-series of microscopic variables in Fourier space of a fluid flow.** Fourier variables  $\tilde{a}_{\eta_1=(1,3,3,3)}$  (top) and  $\tilde{a}_{\eta_2=(1,2,3,4)}$  (bottom) are inferred by using measured variables  $\tilde{a}_\eta$  for  $\eta \in S$  as well as the past time-series data for all the measured variables  $\tilde{a}_\eta$  for  $\eta \in S \cup \{\eta_1, \eta_2\}$ . We can observe that the inferred time-series almost coincide with the actual ones obtained by the direct numerical simulation of the Navier–Stokes equation even after sufficiently large time has passed since the training phase finished. The inference errors in  $l^1$ -norm averaged over  $t - T \in [0, 2000]$  are 1.8% and 3.5% for  $\tilde{a}_{\eta_1}$  and  $\tilde{a}_{\eta_2}$ , respectively.

We consider the absolute value of Fourier variables of velocity  $\mathcal{F}_{[v_\zeta]}(\kappa, t)$  as the representative microscopic variables:

$$a_\eta(t) = |\mathcal{F}_{[v_\zeta]}(\kappa, t)| := \left| \frac{1}{(2\pi)^3} \int_{\mathbb{T}^3} v_\zeta(x, t) e^{-i(\kappa \cdot x)} dx \right|, \quad (1.4.1)$$

## 1. Machine-learning inference of fluid variables

where  $\eta = (\zeta, \kappa) \in S_0 := \{(\zeta, \kappa_1, \kappa_2, \kappa_3) \in \mathbb{Z}^4 \mid \zeta \in \{1, 2, 3\}, \kappa_1, \kappa_2, \kappa_3 \in [-N_0, N_0]\}$ . Since  $v$  is real,  $a_{(\cdot, \kappa_1, \kappa_2, \kappa_3)} = a_{(\cdot, -\kappa_1, -\kappa_2, -\kappa_3)}$ . The reason why we take the absolute value in eq. (1.4.1) is to kill the rotational invariance of a complex variable and to make an inference possible. We choose a chaotic parameter  $\nu = 0.05862$ , and set  $\mathbf{u}(t)$  as the time-series of  $M = 270$  Fourier variables  $\tilde{a}_\eta$ , where  $\eta \in S := \{(2, \pm\kappa_1, \kappa_2, \kappa_3) \in \mathbb{Z}^4 \mid 1 \leq \kappa_1 \leq N_0, \kappa_1 \leq \kappa_2 \leq \kappa_3 \leq \kappa_1 + 4\}$  and each component is taken mod  $N_0$ , that is,

$$\mathbf{u}(t) = (\{\tilde{a}_\eta\}_{\eta \in S})^t.$$

We also set

$$\mathbf{s}(t) = (\tilde{a}_{(1,3,3,3)}, \tilde{a}_{(1,2,3,4)})^t,$$

where  $(1, 3, 3, 3), (1, 2, 3, 4) \notin S$ . Under the set of parameters in TABLE 1.1 (a) we infer the time-series  $\mathbf{s}(t)$ , which is successful for quite a long time (see Fig. 1.1).

The choice of variables to be trained is not very significant in this study, because the attractor does not show a homogeneous isotropic turbulence, and it has less symmetries. We can see from the Poincaré section of the microscopic variables that the flow is not isotropic and indeterminacy in inference due to the continuous symmetry does not appear. However, by training variables with different types of behaviors, we can construct a reservoir model in less computational costs with lower dimension  $N$  of the reservoir system. In fact, we confirmed that we can infer some other fluid variables including both low-frequency and high-frequency variables from some other training variables. We found that an inference of a high-frequency variable tends to be more difficult, maybe because of the stronger intermittency. Remark that  $D_2$  is useful to represent non-local relatively weak interactions among microscopic variables in the partial inference.

## 1.5 Full-inference of macroscopic variables: Energy function and Energy spectrum

We study an energy function as the representative of a macroscopic variable. We set  $\nu = 0.058$  for which the flow is more turbulent than the previous case. However, the complexity of the dynamics is much less than that for a microscopic variable for the same viscosity. This is because the energy function can be thought of as an averaged quantity of many microscopic variables. The energy function  $E_0(k, t)$  for wavenumber  $k \in \mathbb{N}$  is defined by

$$E_0(k, t) := \frac{1}{2} \int_{D_k} \sum_{\zeta=1}^3 |\mathcal{F}_{[v_\zeta]}(\kappa, t)|^2 d\kappa,$$

where  $D_k := \{\kappa \in \mathbb{Z}^3 \mid k - 0.5 \leq |\kappa| < k + 0.5\}$ . See eq. (1.4.1) for the expression of  $\mathcal{F}_{[v_\zeta]}(\kappa, t)$ . In order to get rid of the high-frequency fluctuation, we take the short-time average

$$E(k, t) = \sum_{s=t-99\Delta s}^t E_0(k, s)/100,$$

where  $\Delta s = 0.05$  is the time step of the integration of the Navier–Stokes equation. This helps us to obtain essential low-frequency dynamics of an energy function and infer its time-series with less computational costs with lower dimension  $N$  of the reservoir vectors. The averaged energy function  $E(k, t)$  will be called an energy function hereafter.

In the training phase for  $t \in (0, T]$ ,  $\mathbf{W}_{\text{out}}^*$  and  $\mathbf{c}^*$  are determined by setting

$$\begin{aligned} \mathbf{u}(t) &= (\tilde{E}(1, t), \tilde{E}(2, t), \dots, \tilde{E}(9, t))^t, \\ \mathbf{s}(t) &= (\tilde{E}(1, t), \tilde{E}(2, t), \dots, \tilde{E}(9, t))^t, \end{aligned}$$

and by following the same procedure as the partial-inference. In the inference phase for  $t > T$ , eq.(1.2.1) is written as

$$\mathbf{r}(t + \Delta t) = (1 - \alpha)\mathbf{r}(t) + \alpha \tanh(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{\text{in}}\hat{\mathbf{s}}(t)),$$

## 1. Machine-learning inference of fluid variables

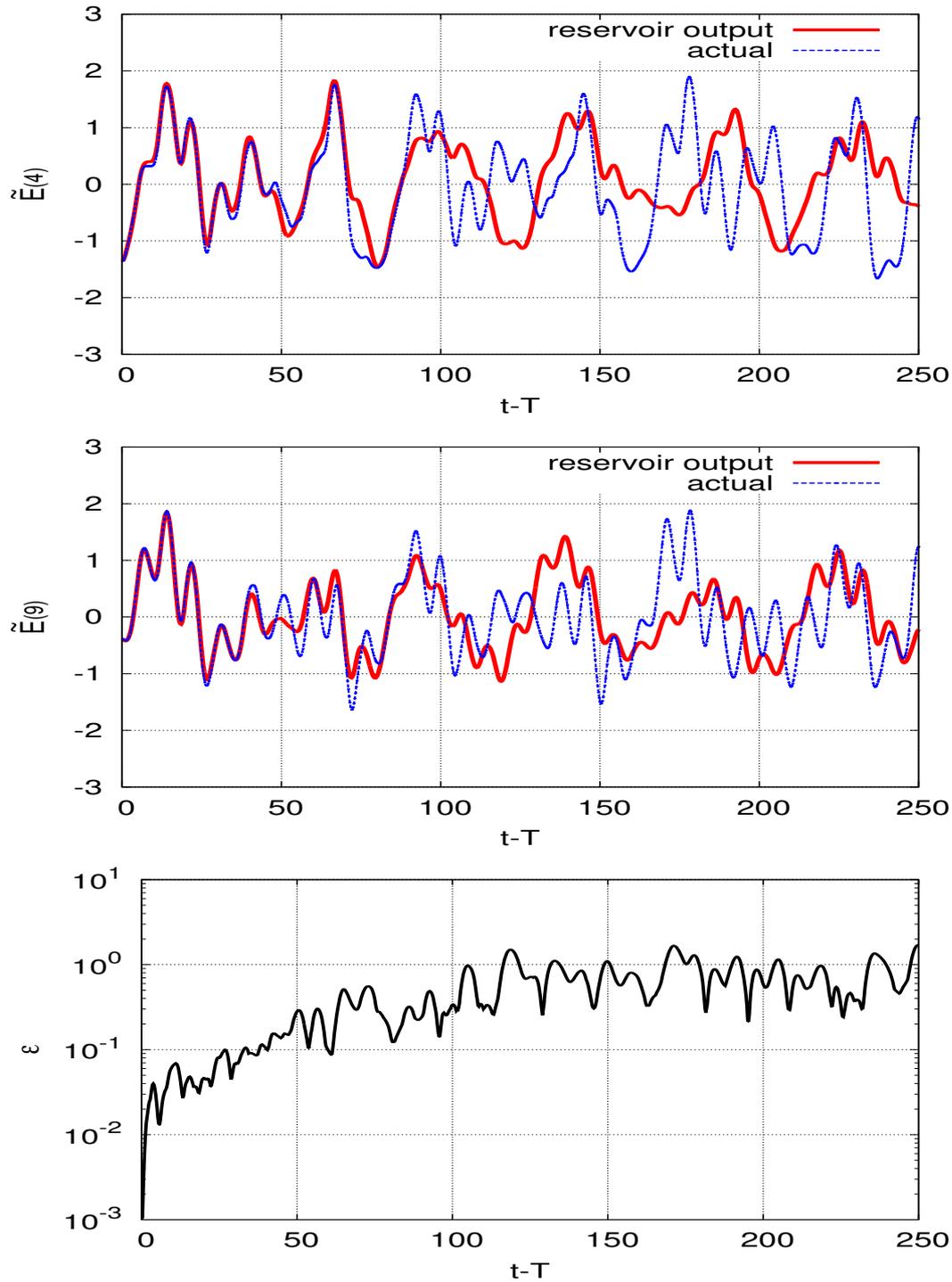


Figure 1.2: **Full-inference of time-series of macroscopic variables of a fluid flow.** Time-series of energy function  $\tilde{E}(k, t)$  for  $k = 4$  (top) and  $9$  (middle) are inferred from the reservoir system in comparison with that of a reference data obtained by the direct numerical simulation of the Navier–Stokes equation. The inference error defined by  $\varepsilon(t) = \sum_{k=1}^{N_0} |\tilde{E}(k, t) - \hat{\tilde{E}}(k, t)| / N_0$  ( $N_0 = 9$ ) is shown to grow exponentially with time up to  $t - T = 100$  (bottom), which is inevitable for a chaotic behavior of a fluid flow. The growth of error within a short time highly depends on the direction of the perturbation vector  $\{\tilde{E}(\cdot, T + \Delta t) - \hat{\tilde{E}}(\cdot, T + \Delta t)\}$ , and its slope can vary in different settings.

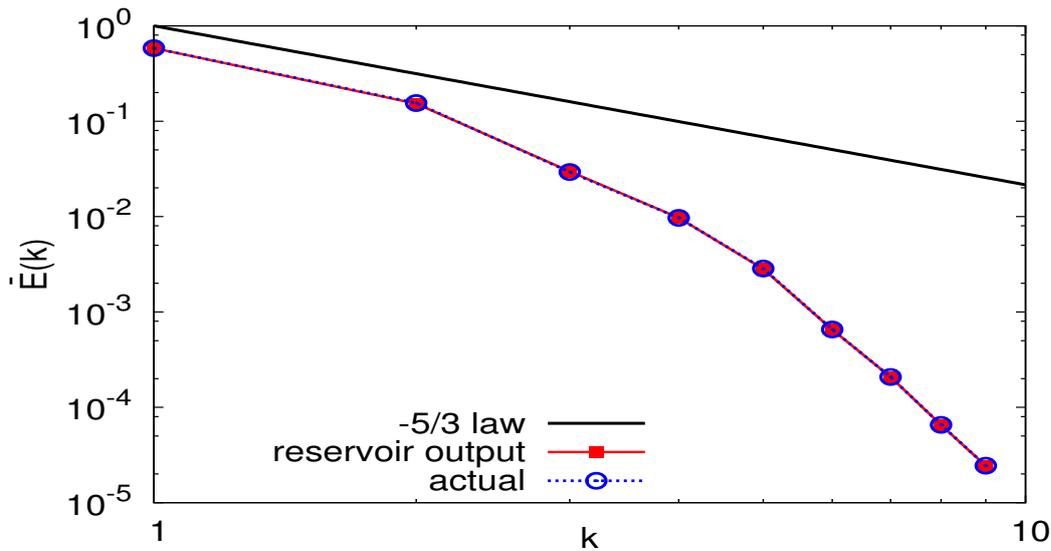


Figure 1.3: **Energy spectrum  $\overline{E}(k)$  reproduced from the reservoir computing.** The spectrum is obtained from the full-inference of an energy function  $E(k, t)$ , which is compared with that for a reference data obtained by the direct numerical simulation of the Navier–Stokes equation. The coincidence of the two energy spectra implies that the reservoir system captures the dynamics of a fluid flow in statistical sense, even after the time-series inference has failed due to the chaotic property (see Fig. 1.2). The Kolmogorov  $-5/3$  law of the energy spectrum is shown as a reference. The relative error of inferred variable  $\hat{E}(k)$  from  $\overline{E}(k)$  ( $k = 1, \dots, 9$ ) is up to 1.3%.

by setting  $\mathbf{u}(t)$  as

$$\hat{\mathbf{s}}(t) = (\hat{E}(1, t), \hat{E}(2, t), \dots, \hat{E}(9, t))^t$$

obtained from eq. (1.2.4). A set of parameters employed here is shown in TABLE 1.1 (b).

We found that an inference of energy functions is successful for some time after finishing training 9-dimensional time-series data of energy functions. The two cases for  $\tilde{E}(4, t)$  and  $\tilde{E}(9, t)$  are shown in Fig. 1.2 (top)(middle). The failure in the long-term time-series inference is inevitable just due to the sensitive dependence on initial condition of a chaotic property of the fluid flow. In fact, the growth rate of error in the energy functions is shown to be exponential for  $t - T \lesssim 100$  in Fig. 1.2 (bottom). However, the energy spectrum  $\overline{E}(k) = \langle E(k, t) \rangle$ , the time average of an energy function  $E(k, t)$ , can be reproduced from the inferred time-series data for  $1000 < t - T < 2000$  (Fig. 1.3). This implies that the reservoir system constructed without the knowledge of microscopic variables captures statistical property correctly, and that the obtained system can be understood as a chaotic dynamical system describing a behavior of energy functions.

## 1.6 Full-inference of Macroscopic variable from only one measurement using delay coordinates

In various experiments and observations of high-dimensional complex phenomena, there are usually much smaller number of measurements than the Lyapunov dimensions of the attractor. Even in such cases we can infer a time-series data by generating high-dimensional input data  $\mathbf{u}$  for the reservoir computation through the delay-coordinate embedding method [41, 37].

Here we exemplify a full-inference of an energy function  $E(4, t)$  for the same flow as in Sec. 1.5, by assuming that the accessible measurement is limited to only one variable  $E(4, t)$  among 9 measurements  $E(k, t)$  ( $k = 1, \dots, 9$ ) used in Sec. 1.5. In order to overcome the lack of sufficiently large number of

## 1. Machine-learning inference of fluid variables

measurements, we introduce 36-dimensional delay-coordinate function with a time delay  $\Delta\tau = 2.5$ , that is,

$$\begin{aligned}\mathbf{u}(t) &= (\tilde{E}(4, t), \tilde{E}(4, t - \Delta\tau), \dots, \tilde{E}(4, t - 35\Delta\tau))^t, \\ \mathbf{s}(t) &= (\tilde{E}(4, t), \tilde{E}(4, t - \Delta\tau), \dots, \tilde{E}(4, t - 35\Delta\tau))^t.\end{aligned}$$

An inferred time-series of  $\tilde{E}(4, t)$  is shown in Fig. 1.4, which is as successful as the case when there are 9 measurements in Fig. 1.2 (top). A set of parameters employed here is shown in TABLE 1.1 (c).

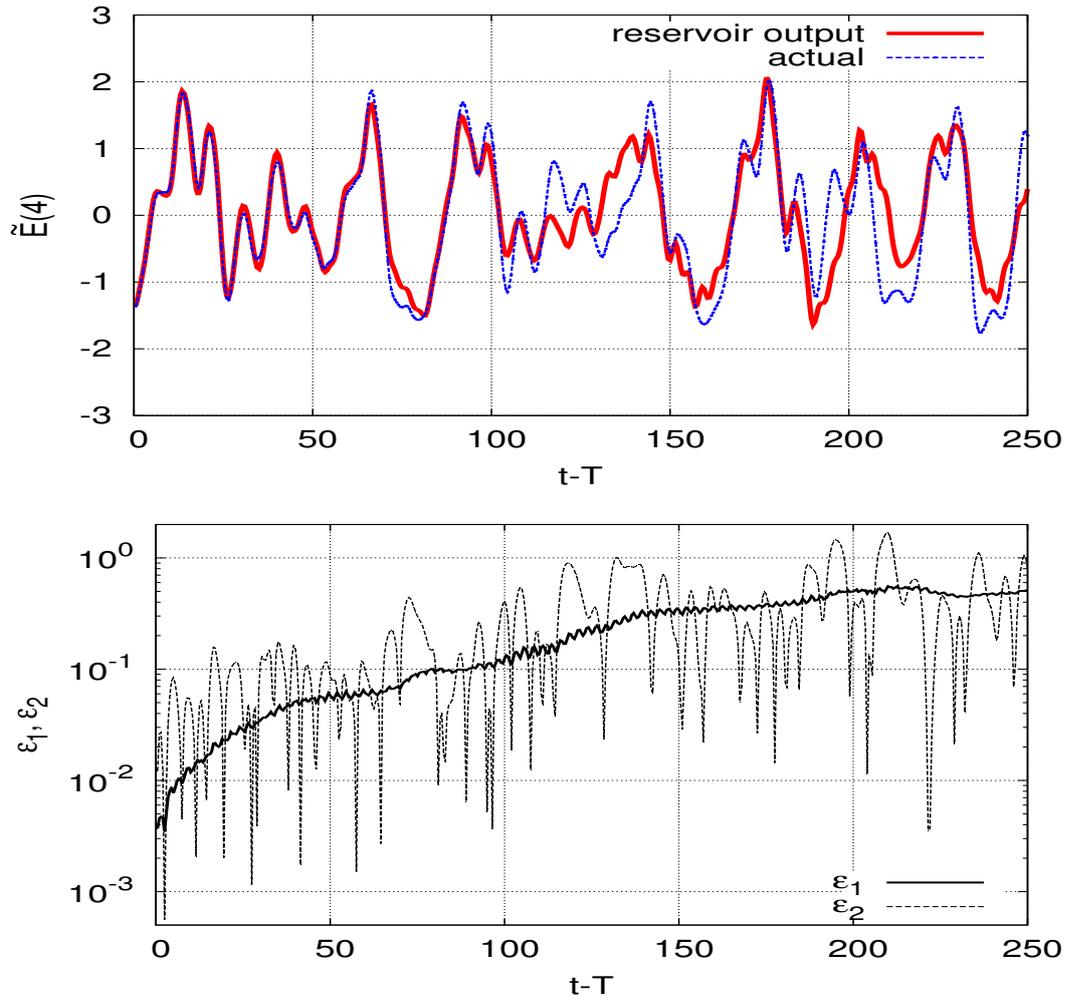


Figure 1.4: **Full-inference of a macroscopic variable using the delay coordinates of only one measurement.** We infer an energy function  $\tilde{E}(4, t)$  for the same time range as in Fig. 1.2 (top) from only one measurement  $\tilde{E}(4, t)$ . The inferred time-series of  $\tilde{E}(4, t)$  is shown together with a reference data obtained by the direct numerical simulation of the Navier–Stokes equation (top). Errors for the inference  $\epsilon_1(t) = \sum_{n=0}^{35} |\tilde{E}(4, t - n\Delta\tau) - \hat{\tilde{E}}(4, t - n\Delta\tau)|^2 / 36$  and  $\epsilon_2(t) = |\tilde{E}(4, t) - \hat{\tilde{E}}(4, t)|$  are shown (bottom).

## 1.7 Discussion and remarks

We have succeeded in inferring time-series of both microscopic and macroscopic variables of a three-dimensional fluid flow by machine-learning technique using reservoir computing. The method is especially useful in generating an arbitrarily long time-series data of macroscopic variables as well as a statistical

## 1. Machine-learning inference of fluid variables

property with small computational costs. That is, in order to generate a time-series data of a macroscopic variable of a fluid flow, we do not need to refer microscopic behaviors. It takes roughly 1/80 of time to obtain a time-series of the energy functions  $E(k)$  with the same time-lengths, when we use the model constructed by the reservoir computation. The Navier–Stokes equation is calculated by 13718-dimensional ODEs with the 4-stage Runge–Kutta method (time step 0.05), whereas the model is calculated by 3200-dimensional map whose iterate corresponds to the time step 0.25.

The difficulty in the construction of a reservoir model can vary mainly depending on the viscosity  $\nu$ . As the degree of turbulence increases by decreasing  $\nu$ , longer training time  $T$  and higher dimension  $N$  of the reservoir state vector  $\mathbf{r} \in \mathbb{R}^N$  are required. However, for macroscopic variables the construction is relatively easy, even when the flow is turbulent. Because the degree of instability of a macroscopic behavior is relatively low in comparison with that of a microscopic behavior.

It is expected that our procedure will work, even if a high-frequency noise is added to the training data, because even in our current computation we have applied a low-pass filter for the inference of macroscopic variables. Although our approach focuses on constructing a model for a fluid flow with a fixed parameter  $\nu$ , it will be very interesting to consider a framework of the construction of a model with a parameter.

When we do numerical computation of the Navier–Stokes equation, we employ some discretized expressions using Fourier spectrum method, finite difference method and finite element method. The obtained reservoir system constructed from data can be understood as one of such expressions, describing a macroscopic (or a microscopic) dynamics of a fluid flow.

It is known that there is a difficulty in obtaining a closed form equation of macroscopic behavior of a fluid flow from the Navier–Stokes equation analytically, so called a “closure problem”. That is, in order to express the dynamics of the  $n$ -th moment variables, the  $n + 1$ -th moment variables are required for any positive integer  $n$ . Our study on the data-driven modeling may give us insights on this kind of problem. For a relatively large value of  $\nu$  considered in this chapter,  $\{E(k)\}_{k=1}^K$  seems to be enough for representing the dynamics of  $E(k)$ , whereas  $\{E(k)\}_{k=1}^K$  will not be enough for more turbulent case with a smaller value of  $\nu$ , even if  $K$  is chosen large enough. In such a case time-delay variables can be used for generating high-dimensional input data as are used in Sec. 1.6

## 1.8 Acknowledgements

We would like to thank anonymous referees for the critical reading of the manuscript and giving us some insightful comments to improve our results. KN was supported by the Leading Graduate Course for Frontiers of Mathematical Sciences and Physics (FMSP) at the University of Tokyo. YS was supported by the JSPS KAKENHI Grant No.17K05360 and JST PRESTO JPMJPR16E5. Part of the computation was supported by the Collaborative Research Program for Young · Women Scientists of ACCMS and IIMC, Kyoto University.

## Chapter 2

# Machine-learning construction of a model for a macroscopic fluid variable using the delay-coordinate of a scalar observable

**Abstract:** We construct a data-driven dynamical system model for a macroscopic variable the Reynolds number of a high-dimensionally chaotic fluid flow by training its scalar time-series data. We use a machine-learning approach, the reservoir computing for the construction of the model, and do not use the knowledge of a physical process of fluid dynamics in its procedure. It is confirmed that an inferred time-series obtained from the model approximates the actual one and that some characteristics of the chaotic invariant set mimic the actual ones. We investigate the appropriate choice of the delay-coordinate, especially the delay-time and the dimension, which enables us to construct a model having a relatively high-dimensional attractor with low computational costs.

### 2.1 Introduction

Reservoir computing is a brain-inspired machine-learning technique that employs a data-driven dynamical system. The framework was proposed as Echo-State Networks [22, 23] and Liquid-State Machines [32], and it has been found to be effective in the inference of a future such as time-series, frequency spectra and the Lyapunov spectra [1, 18, 19, 30, 35, 36, 45].

A reservoir is a recurrent neural network whose internal parameters are not adjusted to fit the data in the training process. Only an output layer is trained. Therefore, the total computational costs are relatively low in comparison with many other machine learning techniques having the same dimensional neural networks. Many physical phenomena including a fluid flow are deterministic, and thus can be described by a high-dimensional dynamical system, even though they have a complex behavior. That is why the reservoir computing with a high-dimensional neural networks can be useful for the construction of a model for such a phenomenon.

In previous chapter [34], we infer both microscopic and macroscopic behaviors of a three-dimensional chaotic fluid flow using reservoir computing. We presented two ways of inference of the complex behavior: the first, called partial inference, requires continued knowledge of partial time-series data during the inference as well as past time-series data, while the second, called full inference, requires only past time-series data as training data. For the first case, we are able to infer long-time motion of microscopic fluid variables. For the second case, we showed that the reservoir dynamics constructed from only past data of energy functions can infer the future behavior of energy functions and reproduced the energy spectrum.

In various experiments and observations of high-dimensional complex phenomena, there are usually much smaller number of measurements than the Lyapunov dimensions of the attractor. Even in such cases, we can efficiently construct a dynamical model by generating high-dimensional input data  $\mathbf{u}$  for the reservoir computing by using the delay-coordinate [34, 37, 41].

The previous chapter focuses on the model construction and the full-inference of a macroscopic variable, the Taylor microscale Reynolds number, when the scalar time-series is accessible as measurements. We evaluate the model in many ways, and discuss details of the appropriate choice of the delay-coordinate

## 2. Machine-learning construction using the delay-coordinate of a scalar observable

created from the single observable. This will be useful for readers who wish to construct a reservoir model by themselves.

After reviewing the procedure of the reservoir computing in Sec. 2.2 and the generation of time-series data of a fluid flow in Sec. 2.3, we show that the constructed reservoir model recovers various properties of a fluid flow obtained from the Navier–Stokes equation in Sec. 2.4. We investigate the effective choice of delay-coordinate in order to construct a model in Sec. 2.5. We summarize our results in Sec. 2.6.

### 2.2 Reservoir computing

Reservoir computing is recently used in the inference of complex dynamics [18, 29, 30, 35, 36]. It focuses on the determination of a linear function from the reservoir state vector to variables to be inferred (see eq. (2.2.5)). Here we review the outline of the method [23, 30]. In this chapter we construct a model dealing with so called full-inference, in which there is no observable data in the inference phase [34].

We consider a dynamical system

$$\frac{d\phi(t)}{dt} = \mathbf{f}(\phi(t)),$$

together with a pair of  $\phi$ -dependent, vector valued variables

$$\mathbf{u}(t) = \mathbf{h}_1(\phi(t)) \in \mathbb{R}^M \quad \text{and} \quad \mathbf{s}(t) = \mathbf{h}_2(\phi(t)) \in \mathbb{R}^M. \quad (2.2.1)$$

We seek a method for using the knowledge of  $\mathbf{u}$  to determine an estimate  $\hat{\mathbf{s}}$  of  $\mathbf{s}$  as a function of time when direct measurement of  $\mathbf{s}$  is not available. We have a knowledge  $\mathbf{u}$  and  $\mathbf{s}$  during the **training phase** for  $t \leq T$ ,  $\mathbf{u}$  and  $\mathbf{s}$  are unknown during the **inference phase** for  $t > T$ . Therefore,  $\mathbf{u}$  during the inference phase is replaced by  $\hat{\mathbf{s}}$  in the previous step. See eq. (2.2.8) for the detail.

The dynamics of the **reservoir state vector**

$$\mathbf{r}(t) \in \mathbb{R}^N \quad (N \gg M),$$

is defined by the neural network

$$\mathbf{r}(t + \Delta t) = (1 - \alpha)\mathbf{r}(t) + \alpha \tanh(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{\text{in}}\mathbf{u}(t)), \quad (2.2.2)$$

where  $\Delta t$  is a relatively short time step, and

$$\tanh(\mathbf{q}) = (\tanh(q_1), \tanh(q_2), \dots, \tanh(q_N))^{\text{T}},$$

for a vector  $\mathbf{q} = (q_1, q_2, \dots, q_N)^{\text{T}}$ . Here,  $\text{T}$  represents the transpose of a matrix. The matrix  $\mathbf{A}$  is a weighted adjacency matrix, and the  $M$ -dimensional input  $\mathbf{u}$  is fed in to the  $N$  reservoir nodes via a linear input weight matrix denoted by  $\mathbf{W}_{\text{in}}$ . The parameter  $\alpha$  ( $0 < \alpha \leq 1$ ) adjusts the nonlinearity of the dynamics of  $\mathbf{r}$ , and is chosen depending upon the complexity of the dynamics of measurements and the time step  $\Delta t$ .

Each row of  $\mathbf{W}_{\text{in}}$  has one nonzero element, chosen from a uniform distribution on  $[-\sigma, \sigma]$ . The matrix  $\mathbf{A}$  is chosen from a sparse random matrix in which the fraction of nonzero matrix elements is  $D/N$ , so that the average degree of a reservoir node is  $D$ . The  $D$  non-zero components are chosen from a uniform distribution on  $[-1, 1]$ . Then we uniformly rescale all the elements of  $\mathbf{A}$  so that the largest value of the magnitudes of its eigenvalues becomes  $\rho$ .

The output, which is a  $M$ -dimensional vector, is taken to be a linear function of the reservoir state vector  $\mathbf{r}$ :

$$\hat{\mathbf{s}}(t) = \mathbf{W}_{\text{out}}\mathbf{r}(t) + \mathbf{c}. \quad (2.2.3)$$

The reservoir state vector  $\mathbf{r}$  evolves following eq. (2.2.2) with input  $\mathbf{u}(t)$ , starting from random initial state  $\mathbf{r}(-T_0)$  whose elements are chosen from  $(0, 1]$  in order not to diverge, where  $T_0 = L_0\Delta t$  ( $\gg 1$ ) is the transient time for  $\mathbf{r}(t)$  ( $t > 0$ ) to be on the attractor. We obtain  $L = T/\Delta t$  steps of reservoir state vectors  $\{\mathbf{r}(l\Delta t)\}_{l=1}^L$  by iterating eq. (2.2.2), while we record the variables  $\{\mathbf{s}(l\Delta t)\}_{l=1}^L$  by using the actual measurements from eq. (2.2.1) for the training phase.

**Determination of  $\mathbf{W}_{\text{out}}$  and  $\mathbf{c}$ .** We determine  $\mathbf{W}_{\text{out}}$  and  $\mathbf{c}$  so that the reservoir output  $\hat{\mathbf{s}}$  (eq. (2.2.3)) approximates the measurement  $\mathbf{s}$  for  $0 < t \leq T$  (training phase), which is a training process in

## 2. Machine-learning construction using the delay-coordinate of a scalar observable

the reservoir computing. We determine them by minimizing the following quadratic form with respect to  $\mathbf{W}_{\text{out}}$  and  $\mathbf{c}$ :

$$\sum_{l=1}^L \|(\mathbf{W}_{\text{out}}\mathbf{r}(l\Delta t) + \mathbf{c}) - \mathbf{s}(l\Delta t)\|^2 + \beta[\text{Tr}(\mathbf{W}_{\text{out}}\mathbf{W}_{\text{out}}^T)], \quad (2.2.4)$$

where  $\|\mathbf{q}\|^2 = \mathbf{q}^T\mathbf{q}$  for a vector  $\mathbf{q}$ , and the second term is a regularization term introduced to avoid overfitting  $\mathbf{W}_{\text{out}}$  for  $\beta \geq 0$ . When the training is successful,  $\hat{\mathbf{s}}(t)$  should approximate the desired unmeasured quantity  $\mathbf{s}(t)$  for  $t > T$  (inference phase). Following eq. (2.2.3), we obtain

$$\hat{\mathbf{s}}(t) = \mathbf{W}_{\text{out}}^*\mathbf{r}(t) + \mathbf{c}^*, \quad (2.2.5)$$

where  $\mathbf{W}_{\text{out}}^*$  and  $\mathbf{c}^*$  denote the solution for the minimizers of the quadratic form (2.2.4):

$$\mathbf{W}_{\text{out}}^* = \delta\mathbf{S}\delta\mathbf{R}^T(\delta\mathbf{R}\delta\mathbf{R}^T + \beta\mathbf{I})^{-1}, \quad (2.2.6)$$

$$\mathbf{c}^* = -[\mathbf{W}_{\text{out}}^*\bar{\mathbf{r}} - \bar{\mathbf{s}}], \quad (2.2.7)$$

where  $\bar{\mathbf{r}} = \sum_{l=1}^L \mathbf{r}(l\Delta t)/L$ ,  $\bar{\mathbf{s}} = \sum_{l=1}^L \mathbf{s}(l\Delta t)/L$ , and  $\mathbf{I}$  is the  $N \times N$  identity matrix,  $\delta\mathbf{R}$  (respectively,  $\delta\mathbf{S}$ ) is the matrix whose  $l$ -th column is  $\mathbf{r}(l\Delta t) - \bar{\mathbf{r}}$  (respectively,  $\mathbf{s}(l\Delta t) - \bar{\mathbf{s}}$ ) (see [31] P.140 and [43] Chapter 1 for details).

In the inference phase for  $t > T$ , eq.(2.2.2) is written as

$$\mathbf{r}(t + \Delta t) = (1 - \alpha)\mathbf{r}(t) + \alpha \tanh(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{\text{in}}\hat{\mathbf{s}}(t)), \quad (2.2.8)$$

by setting  $\mathbf{u}(t)$  as  $\hat{\mathbf{s}}(t)$  obtained from eq. (2.2.5).

We define a reservoir model by eqs. (2.2.5) and (2.2.8) under the values determined by eqs. (2.2.6) and (2.2.7) through the training data in a time-interval  $[0, T]$ . The main variables and matrices in the reservoir computing are summarized in Table 2.1.

**Normalization of a variable.** In order to consider the effect of all the variables equally, we take the normalized value  $\tilde{x}(t)$  for each variable  $x(t)$ , which will be used in the procedure of our reservoir computing:

$$\tilde{x}(t) = [x(t) - X_1]/X_2,$$

where  $X_1$  is the mean value and  $X_2$  is the variance. When we reconstruct  $x(t)$  in the inference phase from  $\tilde{x}(t)$ , we employ  $X_1$  and  $X_2$  obtained in the training phase. Due to the normalization we can avoid adjustments of  $\sigma$ .

**Parameter choice.** We apply a method of reservoir computing described above in order to construct a model. The sets of parameter values used are shown in Table 2.2.

variable	
$\mathbf{u} (\in \mathbf{R}^M)$	input variable
$\mathbf{r} (\in \mathbf{R}^N)$	reservoir state vector
$\mathbf{s} (\in \mathbf{R}^M)$	actual output variable obtained from Navier–Stokes equation
$\hat{\mathbf{s}} (\in \mathbf{R}^M)$	inferred output variable obtained from reservoir computing
$\mathbf{A} (\in \mathbf{R}^{N \times N})$	weighted adjacency matrix
$\mathbf{W}_{\text{in}} (\in \mathbf{R}^{M \times N})$	linear input weight
$\mathbf{W}_{\text{out}} (\in \mathbf{R}^{N \times M})$	matrix used for translation from $\mathbf{r}$ to output variable $\hat{\mathbf{s}}$
$\mathbf{c} (\in \mathbf{R}^M)$	vector used for translation from $\mathbf{r}$ to output variable $\hat{\mathbf{s}}$
$\tilde{x}$	normalized variable of $x$

Table 2.1: **The list of variables and matrices in the reservoir computing.**

## 2.3 Generation of a fluid flow data

Modelling and inference of a fluid flow are important problems in many areas [11, 34]. In this chapter, we construct a model for a macroscopic variable of a fluid flow, especially the time-dependent ‘‘Taylor microscale Reynolds number’’ which reflects the degree of complexity in the fluid flow. We generate

## 2. Machine-learning construction using the delay-coordinate of a scalar observable

parameter		Sec. 2.4	Sec. 2.5
$M$	dimension of input and output variables	14	Table. 2.3
$\Delta\tau$	delay-time of the delay-coordinate	4.0	Table. 2.3
$N$	dimension of reservoir state vector	3000	2000
$D$	parameter of determining $\mathbf{A}$	120	80
$\Delta t$	time step for reservoir dynamics	0.5	
$T_0$	transient time for $\mathbf{r}$ to be converged	3750	
$T$	training time	40000	
$L_0 (= T_0/\Delta t)$	number of iterations for the transient	7500	
$L (= T/\Delta t)$	number of iterations for the training	80000	
$\rho$	maximal eigenvalue of $\mathbf{A}$	0.7	
$\sigma$	scale of input weights in $\mathbf{W}_{\text{in}}$	0.5	
$\alpha$	nonlinearity degree of reservoir dynamics	0.6	
$\beta$	regularization parameter	0.1	

Table 2.2: **The list of parameters and their values used in the reservoir computing in each section.**

training data by the direct numerical simulation of the Navier–Stokes equation, which is also used for the reference data in the inference phase in order to evaluate the constructed reservoir model. It should be remarked that the Navier–Stokes equation and its physical property are not considered at all when constructing a reservoir model.

**Generation of training data.** In order to generate measurements of the reservoir computing, we employ the direct numerical simulation of the incompressible three-dimensional Navier–Stokes equation under periodic boundary conditions:

$$\begin{cases} \partial_t v - \nu \Delta v + (v \cdot \nabla)v + \nabla \pi = f, & \nabla \cdot v = 0, & \mathbb{T}^3 \times (0, \infty), \\ v|_{t=0} = v_0 & \text{with } \nabla \cdot v_0 = 0, & \mathbb{T}^3, \end{cases}$$

where  $\mathbb{T} = [0, 1)$ ,  $\nu > 0$  is a viscosity parameter,  $\pi(x, t)$  is pressure, and  $v(x, t) = (v_1(x, t), v_2(x, t), v_3(x, t))$  is velocity. Throughout this chapter, we set  $\nu = 0.058$ , under which the fluid flow shows an intermittent behavior between laminar and bursting states. See such a behavior in the bottom panel of Fig. 2.1. We use the Fourier spectral method [21] with  $N_0 (= 9)$  modes in each of three directions, meaning that the system is approximated by  $2(2N_0 + 1)^3 (= 13718)$ -dimensional ordinary differential equations (ODEs). The ODEs are integrated by the 4th-order Runge–Kutta scheme, and the forcing is input into the low-frequency variables at each time step so as to preserve the energy of the low-frequency part. See [21, 34] for the details.

**Reynolds number  $R_\lambda$ .** We focus on the time-series of the Taylor microscale Reynolds number, a macroscopic variable representing the degree of complexity of a fluid flow. The total energy  $E(t)$  is defined by

$$E(t) = \sum_{\kappa \in D} \sum_{\zeta=1}^3 (\mathcal{F}_{[v_\zeta]}(\kappa, t))^2,$$

where

$$\mathcal{F}_{[v_\zeta]}(\kappa, t) := \frac{1}{(2\pi)^3} \int_{\mathbb{T}^3} v_\zeta(x, t) e^{-i(\kappa \cdot x)} dx \quad (\zeta = 1, 2, 3),$$

and  $D = \{(\kappa_1, \kappa_2, \kappa_3) \in \mathbb{Z}^3 \mid \kappa_1, \kappa_2, \kappa_3 \in [-9, 9]\}$ . The Taylor microscale Reynolds number  $\tilde{R}_\lambda(t)$  [20] is defined as follows:

$$\tilde{R}_\lambda(t) := \frac{\sqrt{(2/3)E(t)}\lambda}{\nu} = \sqrt{\frac{20E(t)^2}{3\nu\epsilon(t)}},$$

where

$$\epsilon(t) = 2\nu \sum_{\kappa \in D} \sum_{\zeta=1}^3 |\kappa|^2 (\mathcal{F}_{[v_\zeta]}(\kappa, t))^2,$$

## 2. Machine-learning construction using the delay-coordinate of a scalar observable

is the average rate of energy dissipation per unit mass and

$$\lambda = \left( \frac{15\nu(2/3)E(t)}{\epsilon(t)} \right)^{1/2},$$

is the characteristic length of a turbulent fluid flow. The length roughly corresponds to that of an energy input in this study.

In order to get rid of the high-frequency fluctuation, we take the short-time average

$$R_\lambda(t) = \sum_{l=99}^0 \tilde{R}_\lambda(t - l\Delta t^*)/100,$$

where  $\Delta t^* = 0.05$  is the time step of the integration of the Navier–Stokes equation. This helps us to obtain essential low-frequency dynamics of a Reynolds number and construct a model with less computational costs with lower dimension  $N$  of the reservoir state vectors. The averaged Reynolds number  $R_\lambda$  will be called the Reynolds number, and the time-series generated by the direct numerical simulation in the inference phase will be called the “actual data”.

## 2.4 Construction of a model for a macroscopic variable: Reynolds Number

Using the reservoir computing discussed in Sec. 2.2, we construct a model by training a time-series data of the Reynolds number  $R_\lambda$  (see Sec. 2.3) that shows an intermittent behavior between laminar and bursting states. For its purpose a delay-coordinate vector created from a scalar observable is introduced to the input and output variables.

### 2.4.1 Construction of variables based on delay-coordinate

The choice of variables for the reservoir model is significant. Here, we introduce an  $M$ -dimensional delay-coordinate vector of the Reynolds number with a delay-time  $\Delta\tau$  as input and output variables  $\mathbf{u}(t) = (u_1(t), u_2(t), \dots, u_M(t))^T$  and  $\mathbf{s}(t) = (s_1(t), s_2(t), \dots, s_M(t))^T$  in eq. (2.2.1), that is,

$$\mathbf{u}(t) = (\tilde{R}_\lambda(t), \tilde{R}_\lambda(t - \Delta\tau), \dots, \tilde{R}_\lambda(t - (M - 1)\Delta\tau))^T, \quad (2.4.1)$$

$$\mathbf{s}(t) = (\tilde{R}_\lambda(t), \tilde{R}_\lambda(t - \Delta\tau), \dots, \tilde{R}_\lambda(t - (M - 1)\Delta\tau))^T. \quad (2.4.2)$$

The appropriate choice of the **dimension**  $M$  and the **delay-time**  $\Delta\tau$  of the delay-coordinate will be discussed in Sec. 2.5.

**Determination of a model.** Under the parameters listed in Table 2.2 and randomly chosen matrices  $\mathbf{A}$  and  $\mathbf{W}_{\text{in}}$ , we find a candidate of a reservoir model by fixing  $\mathbf{W}_{\text{out}}^*$  and  $\mathbf{c}^*$  following the procedure explained in Sec 2.2. If the candidate passes a certain criteria concerning the short time inference, the candidate is considered as a model. See Sec. 2.5 for the details of the criteria. Remark that although we can use a training data as some delay components of input data when  $t - (M - 1)\Delta\tau < T$ , we do not use any training data in the inference phase. Hereafter throughout this section, we choose one of the models, and fix the corresponding set of values  $\mathbf{A}$ ,  $\mathbf{W}_{\text{in}}$ ,  $\mathbf{W}_{\text{out}}^*$  and  $\mathbf{c}^*$ .

### 2.4.2 Evaluation of the model

We evaluate the constructed reservoir model for the Reynolds number from several points of view by comparing its property with that of the actual data obtained from the direct numerical simulation of Navier–Stokes equation.

**Time-series.** We confirm that an inference of a time-series of the Reynolds number  $s_1 = \tilde{R}_\lambda$  is successful for some time after finishing the training phase. The time-series of the inferred variable  $\hat{s}_1 = \hat{\tilde{R}}_\lambda(t)$  ( $t > T$ ) is shown with the actual data  $s_1 = \tilde{R}_\lambda(t)$  obtained from the direct numerical simulation of Navier–Stokes equation in the top left panel of Fig. 2.1. The failure in the long-term time-series inference is inevitable just due to the sensitive dependence on the initial condition of a chaotic property of the fluid flow. The two types of errors between the inferred value and the actual one are shown in the top right panel of Fig.

## 2. Machine-learning construction using the delay-coordinate of a scalar observable

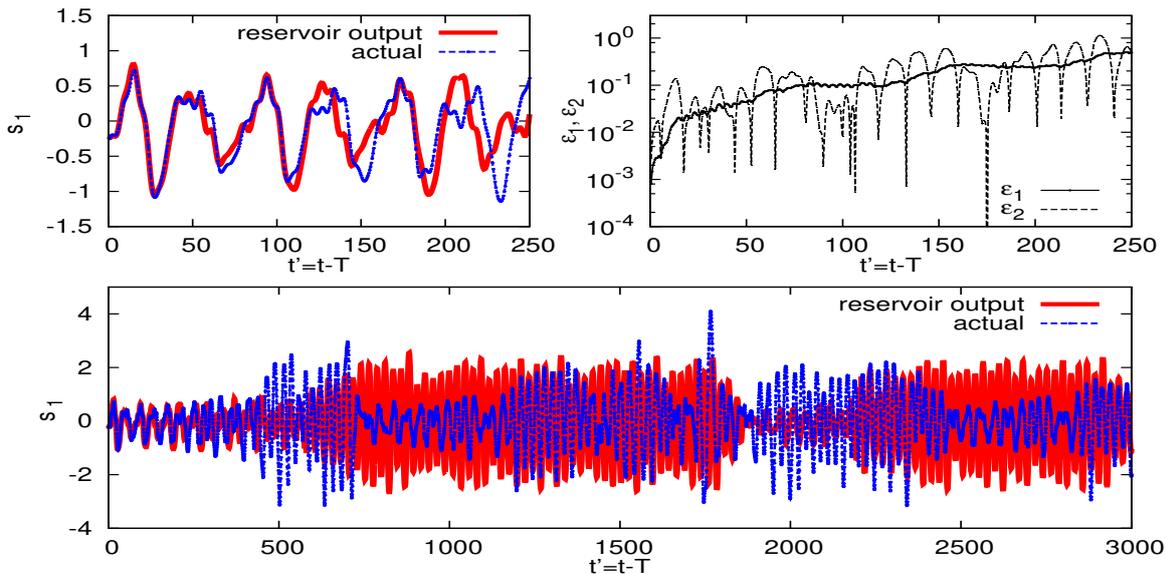


Figure 2.1: **Inference of a time-series of the Reynolds number of a fluid flow.** Time-series of  $s_1 = \tilde{R}_\lambda$  is inferred from the reservoir model in comparison with that of a reference data obtained by the direct numerical simulation of the Navier–Stokes equation (top left). The variable  $t'$  ( $= t - T > 0$ ) denotes the time after finishing the training phase at  $t = T$ . The inference errors  $\varepsilon_1, \varepsilon_2$  defined by  $\varepsilon_1(t) = |\mathbf{s}(t) - \hat{\mathbf{s}}(t)|$ , and  $\varepsilon_2(t) = |s_1(t) - \hat{s}_1(t)| = |\tilde{R}_\lambda(t) - \hat{\tilde{R}}_\lambda(t)|$  are shown to increase exponentially due to the chaotic property (top right). In the bottom figure switching between laminar state with a small amplitude fluctuation and bursting state with a large amplitude fluctuation appear in an inferred time-series of  $s_1 = \tilde{R}_\lambda$ , which are observed in the actual time-series.

2.1. Moreover, the long-time behavior of  $\hat{\tilde{R}}_\lambda(t)$  is shown in the bottom panel, which has qualitatively similar intermittent behaviors to the actual one, intermittent switching between the state of low amplitude fluctuations (laminar state) and the state of high amplitude fluctuations (burst state). Remark that the model trajectory shows a chaotic behavior, but after a long transient it will diverge eventually around  $t' \approx 290000$ .

**Delay Property.** As we employ the delay-coordinate vector for input and output variables of the reservoir computing (eqs. (2.4.1),(2.4.2)), the relation  $s_1(t) = s_m(t + (m - 1)\Delta\tau)$  holds for any  $m$  ( $m = 2, \dots, M$ ) during the training phase. The corresponding relation should also be satisfied in the inference phase. We show the time-series  $\hat{s}_1(t)$  and  $\hat{s}_{14}(t + 13\Delta\tau)$  in Fig. 2.2, which satisfies the relation  $\hat{s}_1(t) \approx \hat{s}_{14}(t + 13\Delta\tau)$ . We can confirm that for almost all  $t$  the relation  $\hat{s}_1(t) \approx \hat{s}_m(t + (m - 1)\Delta\tau)$  is satisfied for any  $m$ . The results imply that our reservoir computing successfully learns the delay property only through training such data.

**Poincaré plane.** We investigate the chaotic set computed from a model trajectory to see whether the inferred chaotic set mimics the actual one. For its purpose we describe the Poincaré plane in comparison with that computed from a trajectory of the direct numerical simulation of the Navier–Stokes equation with the same time length in the Fig. 2.3. The figure suggests that each of the chaotic set is hyperchaotic, that is the dimension of the unstable manifold is two or higher. Although the sections are similar to each other, they are not very close to each other. This may be because the length of the intermittent trajectory is not enough to cover various regions especially in the bursting state.

**Distribution.** Density distributions computed from two inferred trajectories  $\{\hat{s}_1(t)\}$  and those from two actual trajectories  $\{s_1(t)\}$  are shown in Fig. 2.4. We can observe that the distributions computed from trajectories of time lengths 5000 are fluctuating, but the inferred distributions seem to have similar properties to the actual distributions. Relatively large fluctuations in distributions for  $|\hat{s}_1(t)| > 1$  should be due to the intermittency.

**The reservoir model can be used to infer time-series of another time-interval.** We obtained a model just by training the data and it enables us to infer short-time behavior, the shape of an attractor

## 2. Machine-learning construction using the delay-coordinate of a scalar observable

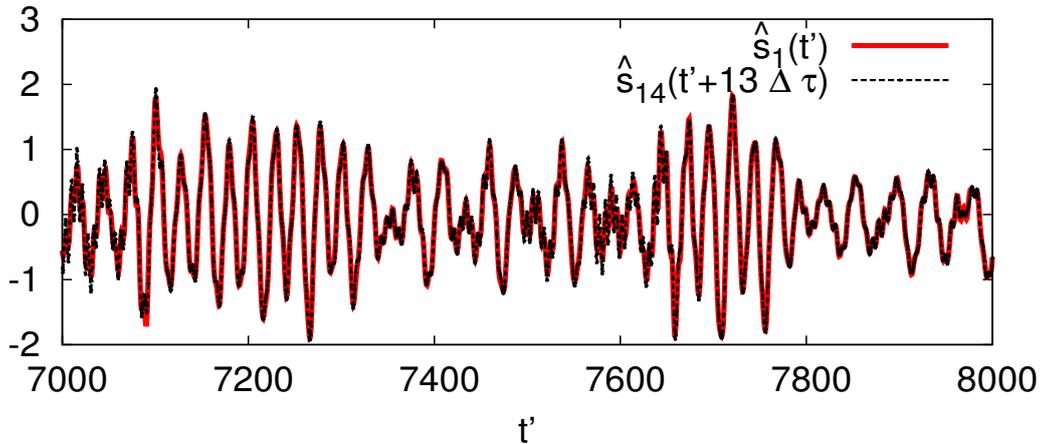


Figure 2.2: **Reproducing the delay property which is to be satisfied for the successfully inferred time-series  $\hat{\mathbf{s}}$ .** We observe that for all values of  $m = 2, \dots, 14$  and for most  $t'$ ,  $\hat{s}_1(t') \approx \hat{s}_m(t' + (m - 1)\Delta\tau)$ , although the time-series of only  $\hat{s}_1(t')$  and  $\hat{s}_{14}(t' + 13\Delta\tau)$  ( $7000 \leq t' \leq 8000$ ) are shown.

and the density distribution. Here we confirm that the model constructed using a certain training data has the ability to infer a short-time behavior of the Reynolds number for the totally different time-interval. In Fig. 2.5, the inferred time-series is shown in comparison with the actual one. For this inference we use the same reservoir model as is used in Fig. 2.1. This means that the reservoir model constructed using the training data at a certain time-interval can become the model for another time-interval. This figure supports the accuracy of the constructed reservoir model. In Fig. 2.6, by using the same model the inference of time-series of the Reynolds number in many different time intervals are shown. For each time-interval, we confirm that the short time inference is successful. This implies that the obtained model can describe the dynamics of the Reynolds number. Note that the time-interval for the successful inference is limited to a relatively short time especially for the bursting phase with large fluctuations maybe due to the high instability.

It should be remarked that the model satisfies the Echo State Property [22], that is, for a given  $\mathbf{u}$ , independent of the initial conditions the time-series of  $\mathbf{r}$  become almost the same (less than  $10^{-15}$  difference in double precision) after a certain transient time (greater than around 100). Therefore, the obtained reservoir model is considered to be a closed form of  $\mathbf{u}$ .

2. Machine-learning construction using the delay-coordinate of a scalar observable

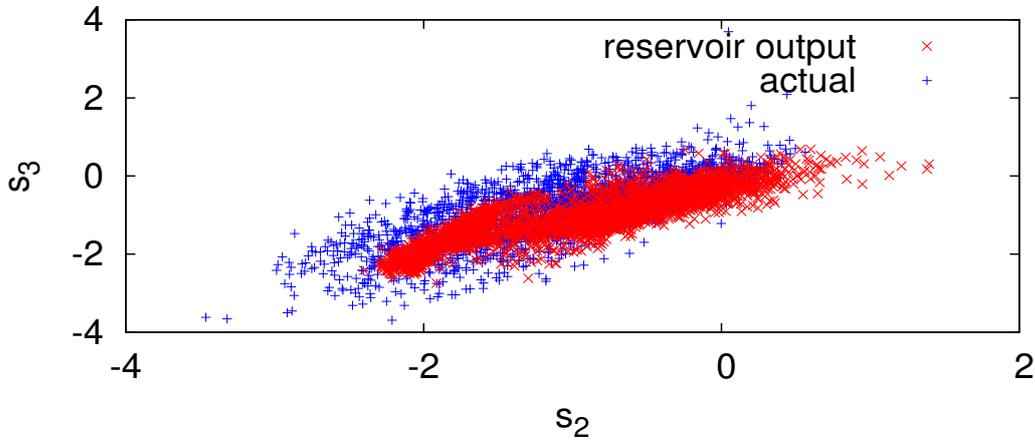


Figure 2.3: Poincaré points on the plane  $(s_2, s_3)$  along the trajectory  $\hat{s}$  obtained from the reservoir model (red) and  $s$  from the Navier–Stokes equation (blue). The time length of each trajectory is 90000. The Poincaré section is defined by  $s_1 = 0$ ,  $ds_1/dt > 0$ . Two sections are similar to each other, although a trajectory generated from the reservoir model does not cover some region of bursting states.

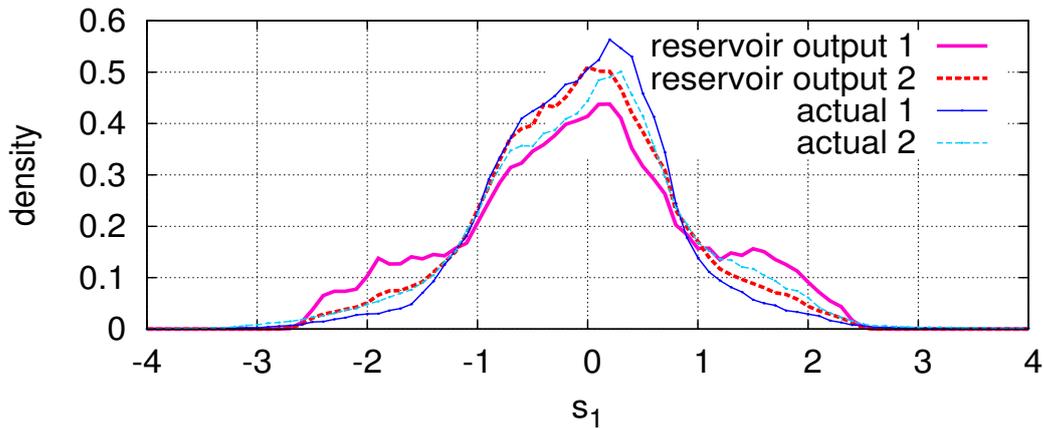


Figure 2.4: Density distributions generated from trajectories for a variable  $s_1$  obtained from the constructed reservoir model (reservoir output) and from the direct numerical simulation of the Navier–Stokes equation (actual). Each trajectory with a time-length 50000 has a different initial condition. The distributions are similar to each other in the sense that the peak is taken at  $s_1 \approx 0.2$ , and the distribution has relatively long tails.

## 2. Machine-learning construction using the delay-coordinate of a scalar observable

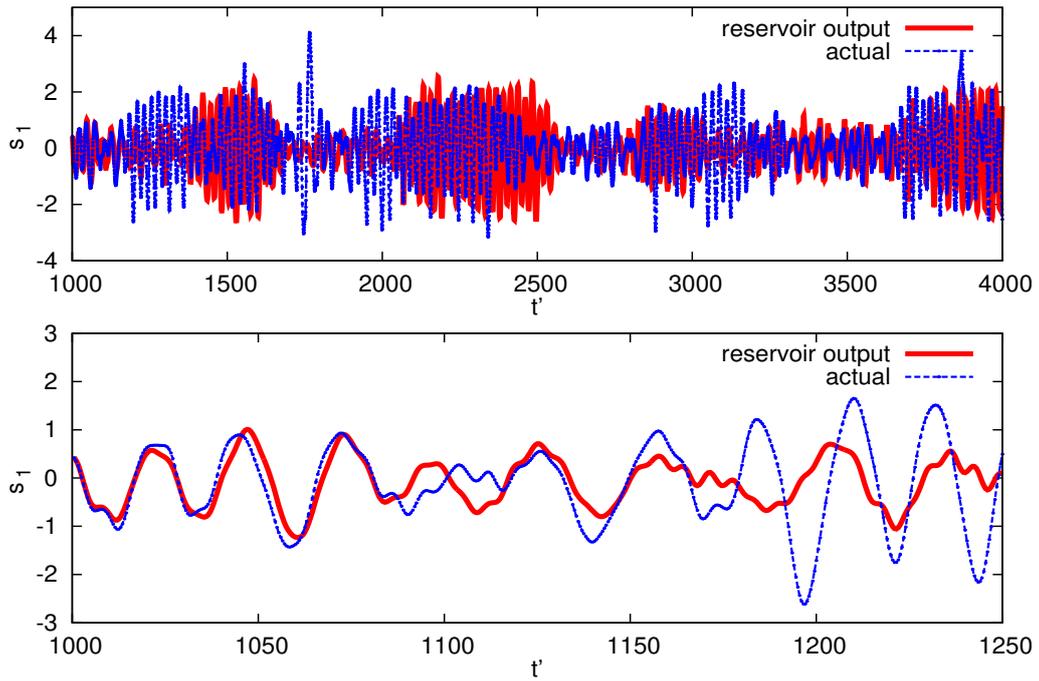


Figure 2.5: **Inference of a time-series of the Reynolds number for  $t' > T_{\text{out}}$  ( $T_{\text{out}} = 1000$ ) using the reservoir model constructed by using the training data for  $t' \leq 0$  (see Fig. 2.1).** We use the same  $\mathbf{W}_{\text{in}}$ ,  $\mathbf{A}$ ,  $\mathbf{W}_{\text{out}}^*$  and  $\mathbf{c}^*$  as those used for the model inferring the trajectory in Fig. 2.1. But we use the time-series  $s_1(t')$  for  $T_{\text{out}} - T_1 < t' < T_{\text{out}}$  as an initial condition, where  $T_1$  is the transient time for the reservoir state vector  $\mathbf{r}(t)$  to be converged. In the top panel, switching between laminar and bursting states is observed in the inferred trajectory. The bottom panel is the enlargement of the top panel, and shows that the model has a predictability for  $1000 < t' < 1080$ .

## 2. Machine-learning construction using the delay-coordinate of a scalar observable

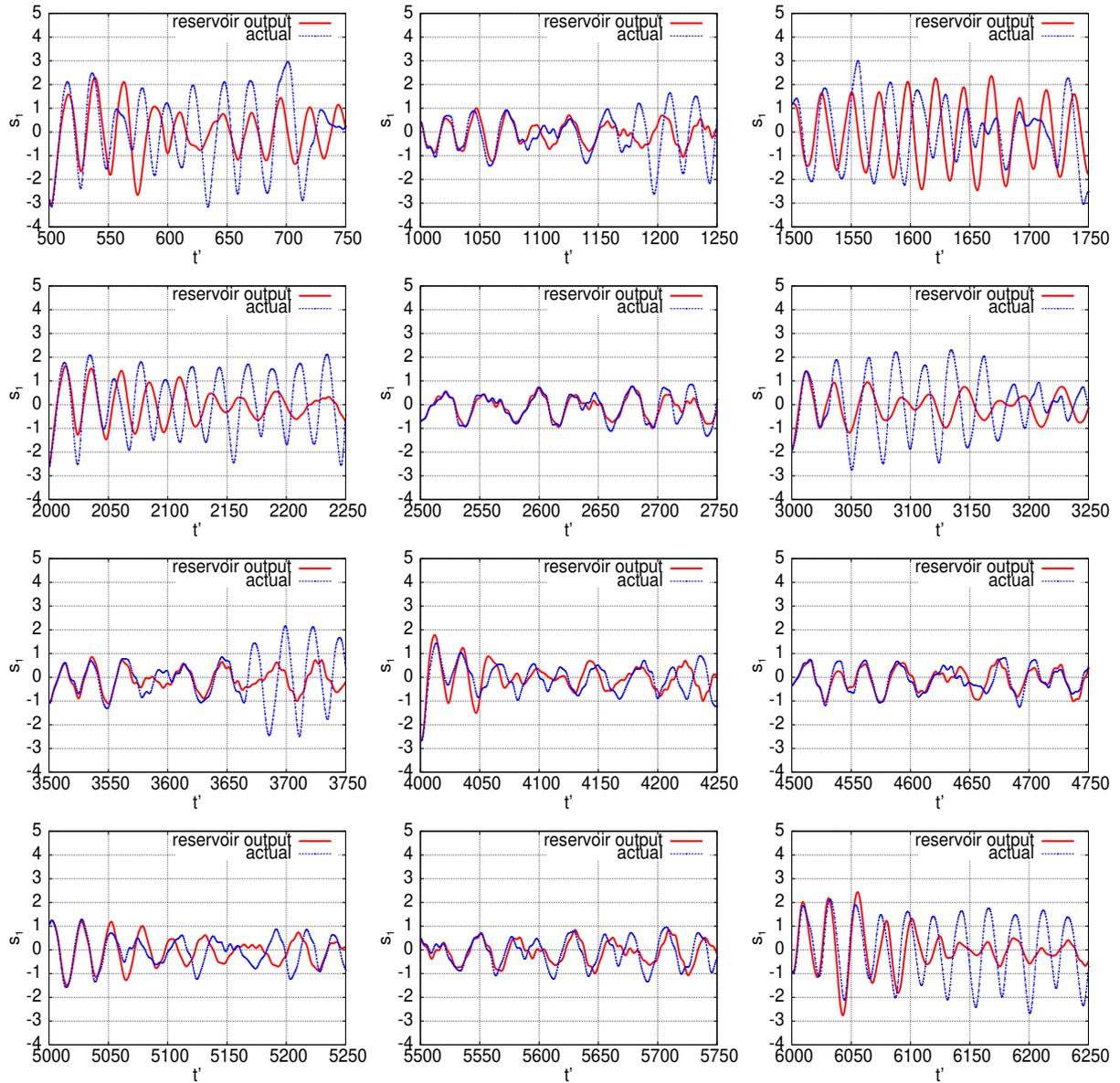


Figure 2.6: **Inference of time-series of the Reynolds number in many time-intervals  $T_{\text{out}} < t' < T_{\text{out}} + 250$  ( $T_{\text{out}} = 500, 1000, \dots, 6000$ ) using the same reservoir model constructed by using the training data for  $t' \leq 0$  (see Fig. 2.1 and 2.5.)** As in Fig. 2.5, we only change the initial condition for each case, while the model is fixed after the appropriate choice of  $\mathbf{W}_{\text{in}}, \mathbf{A}, \mathbf{W}_{\text{out}}^*$  and  $\mathbf{c}^*$  is determined by using the training data for  $t' < 0$ .

## 2.5 Choice of delay-coordinate.

We use an  $M$ -dimensional delay-coordinate vector with a delay-time  $\Delta\tau$  (eqs. (2.4.1),(2.4.2)) as input and output variables  $\mathbf{u}$  and  $\mathbf{s}$  in eq. (2.2.1). In this section we investigate the appropriate choice of time-delay  $\Delta\tau$  and the dimension  $M$ .

**Time-correlation.** The auto-correlation function  $C(x)$  along a trajectory  $\{R_\lambda(t)\}$  with respect to the time-difference  $x$  is computed by

$$C(x) = \frac{\frac{1}{J} \sum_{j=0}^{J-1} (R_\lambda(t_0 + j\Delta t^*) - \bar{R}_\lambda)(R_\lambda(t_0 + j\Delta t^* + x) - \bar{R}_\lambda)}{\sqrt{\frac{1}{J} \sum_{j=0}^{J-1} (R_\lambda(t_0 + j\Delta t^*) - \bar{R}_\lambda)^2} \sqrt{\frac{1}{J} \sum_{j=0}^{J-1} (R_\lambda(t_0 + j\Delta t^* + x) - \bar{R}_\lambda)^2}}, \quad (2.5.1)$$

where  $\bar{R}_\lambda$  is the time average of  $R_\lambda(t)$ ,  $\Delta t^*$  is the time step of the discrete trajectory, and  $t_0$  is an initial time of a trajectory. In Fig. 2.7, we show the auto-correlation function  $C(x)$  for a trajectory  $\{R_\lambda(t)\}$  with respect to the delay-time  $x$ . It is observed from Fig. 2.7 (right) that as  $x$  increases from 0,  $C(x)$

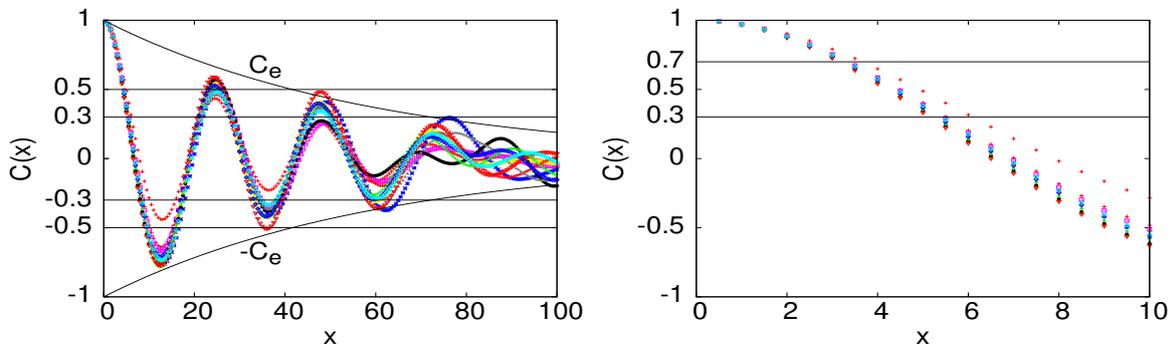


Figure 2.7: **Auto-correlation function  $C(x)$  for a trajectory  $\{R_\lambda(t)\}$  with respect to the value of time-delay  $x$  (left), and its enlarged figure (right).** Auto-correlation function  $C(x)$  is shown together with the straight lines  $\pm 0.3, \pm 0.5$  (left panel), and  $0.3, 0.7$  (right panel). Each of the different colors represents  $C(x)$  computed from a trajectory from a different initial condition with time-lengths 5000. The difference is mainly due to the intermittent property of the dynamics. In the left panel the envelope  $C_e(x)(= \exp(-x/60))$  is shown to go below 0.5 when  $x \approx 40$ , and also go below 0.3 when  $x \approx 75$ . From the right panel  $C(x)$  is shown to go below 0.7 at the first time, when  $x \approx 3.0$ , and go below 0.3 at the first time, when  $x \approx 5.0$ .

goes below 0.7 and 0.3 when  $x \approx 3.0$  and  $5.0$ , respectively.

The observation suggests that the value of the delay-time  $\Delta\tau$  is to be chosen around 3.0-5.0. If  $\Delta\tau < 3.0$ , the consecutive two components of a delay-coordinate vector in (2.4.1),  $R_\lambda(t)$  and  $R_\lambda(t - \Delta\tau)$  behave too similarly, and if  $\Delta\tau > 5.0$ , the consecutive two components behave too differently, and some dynamics to be captured may be missing.

**Delay-time and dimensions.** Based on the above implication about the auto-correlation function in Fig. 2.7, we investigate the effective delay-time  $\Delta\tau$  and dimensions  $M$ . We focus on the delay-time  $\Delta\tau \approx 3.0$ -5.0 in Table 2.3. We infer time-series of the Reynolds number  $R_\lambda(t)$  (actually its normalized value  $\tilde{R}_\lambda(t)$ ) using the procedure in Sec. 2.2 by employing the delay-coordinate in eq. (2.5). We tried 8160 cases for each set of parameters  $(\Delta\tau, M)$ , for which matrices  $\mathbf{A}$  and  $\mathbf{W}_{\text{in}}$  are chosen randomly, and the number of successful cases are counted in TABLE 2.3. We say that the inference of  $s_1(t')$  ( $t' = t - T > 0$ )

## 2. Machine-learning construction using the delay-coordinate of a scalar observable

(a)  $(e_{60}, e_{90}) = (0.14, 0.30)$

$\Delta\tau \setminus M$	10	11	12	13	14	15	16	17	18	19	20
3.0	0	0	0	0	0	1	19	24	<b>43</b>	37	27
3.5	0	0	0	11	20	28	<b>57</b>	48	21	11	7
4.0	0	3	18	43	<u><b>107</b></u>	59	21	14	2	4	5
4.5	3	14	43	<b>54</b>	21	15	8	1	1	1	0
5.0	10	24	<b>26</b>	19	9	1	1	1	0	0	0

(b)  $(e_{60}, e_{90}) = (0.13, 0.17)$

$\Delta\tau \setminus M$	10	11	12	13	14	15	16	17	18	19	20
3.0	0	0	0	0	0	0	3	6	<b>10</b>	8	4
3.5	0	0	0	2	3	5	<b>6</b>	4	1	3	1
4.0	0	0	2	8	<u><b>14</b></u>	10	1	4	1	0	1
4.5	1	1	8	<u><b>14</b></u>	1	0	1	0	0	0	0
5.0	2	4	<b>6</b>	<b>6</b>	3	0	1	0	0	0	0

Table 2.3: **The number of successful trials for each choice of the delay-time  $\Delta\tau$  and the dimension  $M$  of the delay-coordinate.** The matrices  $\mathbf{A}$  and  $\mathbf{W}_{\text{in}}$  are chosen randomly, and the number of successful cases are counted. See Table. 2.2 for the parameter values. We say the inference is successful, if the three conditions (i)(ii)(iii) in (2.5.2) hold, where the criteria  $(e_{60}, e_{90})$  are set as (a)(0.14, 0.30) and (b)(0.13, 0.17). For each set of values  $(\Delta\tau, M)$  we tried 8160 cases of  $\mathbf{A}$  and  $\mathbf{W}_{\text{in}}$ . For each value of  $\Delta\tau$ , the best choice of  $M$  is identified by the bold number(s) (blue), and the best among each criterion is identified by the underlined bold number(s) (red).

is successful if the conditions

$$\begin{cases} \text{(i) the time average along } |\hat{s}_1(t')| < 3 \text{ for } t' \leq 3000, \\ \text{(ii) the error } \varepsilon_2(t') = |s_1(t') - \hat{s}_1(t')| = |\tilde{R}_\lambda(t') - \hat{R}_\lambda(t')| < e_{60} \text{ for all } t' \leq 60, \\ \text{(iii) the error } \varepsilon_2(t') < e_{90} \text{ for all } t' \leq 90, \end{cases} \quad (2.5.2)$$

hold, where the criteria  $(e_{60}, e_{90})$  are set as (a)(0.14, 0.30) and (b)(0.13, 0.17). Remark that the condition (i) is given so as to get rid of a candidate which diverges within a short time, as  $|s_1(t')| < 3$  for almost all  $t$  even in the bursting region. For each case we use the same training data and the starting time of the inference as in Fig. 2.1.

It is observed that the delay-time  $\Delta\tau$  and the dimension  $M$  of the delay-coordinate are chosen so that  $\Delta\tau \approx 4.0$ -4.5, and  $M\Delta\tau \approx 55$ -60, which correspond to  $C(\Delta\tau) \approx 0.45$ -0.55 and its envelope  $C_e(M\Delta\tau) \approx 0.35$ -0.40, respectively (see the left panel of Fig. 2.7 for the envelope  $C_e$ ). For  $\Delta\tau = 4.0$  and  $M = 14, 15$  by computing 16 times more cases, we confirmed that the rate of successful trials does not change much. In addition, even when we change the value of  $N$  such as 1000 or 3000, we obtain almost the same results.

## 2.6 Summary and discussion

By training a time-series data of a macroscopic quantity the Reynolds number of a fluid flow, we construct a closed form system describing its behavior without the knowledge of a physical process. We evaluate the obtained model in many ways. In particular, the model is confirmed to have a time-series predictability in many time intervals.

In order to construct a model from a scalar time-series data, we introduce a time-delay coordinate. From our investigations, the time-delay should be chosen to be the lowest value  $\Delta\tau (> 0)$  so that the auto-correlation function  $C$  is  $0.45 < C(\Delta\tau) < 0.55$  at the first time, and that the dimension  $M$  of the delay-coordinate should be chosen so that the envelope  $C_e$  of the auto-correlation function  $C$  is  $0.35 < C_e(M\Delta\tau) < 0.40$ .

It should be remarked that the obtained reservoir model has a chaotic set on which a trajectory approximates the actual one, but the set is not an attractor. This may be due to the lack of training data, especially in the bursting state. The clarification is remained as a future study.

## 2.7 Acknowledgements

KN was supported by the Leading Graduate Course for Frontiers of Mathematical Sciences and Physics (FMSP) at the University of Tokyo and grant-in-aid for JSPS fellow 19J12482. YS was supported by the JSPS KAKENHI Grant No.17K05360 and JST PRESTO JPMJPR16E5. Part of the computation was supported by the Collaborative Research Program for Young · Women Scientists of ACCMS and IIMC, Kyoto University and by “High Performance Computing Infrastructure” in Japan (Project ID: jh190070).

## Chapter 3

# Direction of Vorticity and a Refined Regularity Criterion for the Navier–Stokes Equations with Fractional Laplacian

**Abstract:** We give a refined regularity criterion for solutions of the three-dimensional Navier–Stokes equations with fractional dissipative term  $(-\Delta)^{\alpha/2}v$ . The criterion is composed of the direction field of the vorticity and its magnitude simultaneously. Our result is a generalized of previous results by H. Beirão da Veiga and L. Berselli (2002), and Y. Zhou (2005). Moreover, our result mentioned about the relation between the solution of the Navier–Stokes equations and the Euler equations.

### 3.1 Introduction

We consider the three-dimensional incompressible Navier–Stokes equations with fractional powers of the Laplacian:

$$\partial_t v + (v \cdot \nabla)v = -\nabla\pi - (-\Delta)^{\alpha/2}v \quad \text{in } \mathbb{R}^3 \times (0, \infty), \quad (3.1.1)$$

$$\operatorname{div} v = 0 \quad \text{in } \mathbb{R}^3 \times (0, \infty), \quad (3.1.2)$$

$$v(x, 0) = v_0(x) \quad \text{in } \mathbb{R}^3, \quad (3.1.3)$$

where  $v = v(x, t) = (v_1(x, t), v_2(x, t), v_3(x, t))$  is the velocity of the fluid flows,  $\pi = \pi(x, t)$  is the pressure, and  $v_0(x)$  is a given initial velocity field satisfying  $\operatorname{div} v_0 = 0$ . Furthermore, a general fractional Laplacian  $(-\Delta)^{\alpha/2}$  ( $\alpha > 0$ ) is defined by

$$\mathcal{F} [(-\Delta)^{\alpha/2}f](\xi) = |\xi|^\alpha \mathcal{F}_{[f]}(\xi),$$

where  $\mathcal{F}_{[f]}$  is the Fourier transform of  $f$ . We denote the equations (3.1.1)–(3.1.3) by  $(\text{NS})_\alpha$ . When  $\alpha = 2$ , the  $(\text{NS})_\alpha$  reduce to the usual Navier–Stokes equations. In this chapter we are concerned with the case  $0 < \alpha \leq 2$ .

The  $(\text{NS})_\alpha$  were first considered by J.L. Lions [28]. It is known in [48] that when  $\alpha > 0$ ,  $(\text{NS})_\alpha$  with  $v_0 \in L^2$  possess a global weak solution. By a weak solution, we mean  $(v, \pi)$  satisfies  $(\text{NS})_\alpha$  in the distribution sense. In addition, we have the basic regularity for the weak solution:

$$v \in L^\infty(0, T; L^2) \cap L^2(0, T; H^{\alpha/2})$$

for any  $T > 0$ . Moreover, N.H. Katz and N. Pavlović [25] showed that if  $2 < \alpha < 5/2$ , the Hausdorff dimension of the singular set at the time of first possible blow-up is at most  $5 - 2\alpha$ . Furthermore, if  $\alpha \geq 5/2$ ,  $(\text{NS})_\alpha$  have a unique global smooth solution [28]. Also, the existence of time global solution under  $\alpha \geq 5/2$  is recovered by Y. Zhou [51].

In [7, 13], the authors claim that the vorticity  $\omega$  which is defined by  $\operatorname{rot} v$  plays an important role in the regularity conditions for the Navier–Stokes equations, where  $\operatorname{rot}$  is a differential operator. Taking the

### 3. Refined Regularity Criterion

operator rot of (3.1.1), we obtain an equation:

$$\partial_t \omega + (-\Delta)^{\alpha/2} \omega = (\omega \cdot \nabla) v - (v \cdot \nabla) \omega. \quad (3.1.4)$$

The velocity can be written in terms of the vorticity through the Biot-Savart law:

$$v(x, t) = \frac{-1}{4\pi} \int_{\mathbb{R}^3} \nabla_y \left( \frac{1}{|x-y|} \right) \times \omega(y, t) dy, \quad (3.1.5)$$

which follows from (3.1.2) provided vorticity decay near infinity. D. Chae, J. Fan and T. Ozawa proved the following theorem on a solution to  $(NS)_\alpha$  in the class

$$C([0, T]; H^s) \cap C^1([0, T]; H^{s-1}) =: E_s(T) \quad s > 5/2,$$

which we call a strong solution in the class  $E_s(T)$ ;

**Theorem 3.1.1** ([6, 12]). Let  $s > 5/2, T > 0$ , and  $\alpha \in (0, 5/2]$ . Let  $v$  be a strong solution of  $(NS)_\alpha$  in the class  $E_s(T)$ . If the vorticity  $\omega(x, t)$  satisfies

$$\omega \in L^p(0, T; L^q), \quad \frac{\alpha}{p} + \frac{3}{q} \leq \alpha, \quad (3.1.6)$$

where  $6/\alpha < q \leq \infty$ . Then  $v$  can be continued to the strong solution in the class  $E_s(T')$  for some  $T < T'$ .

Let  $\xi(x, t)$  be the direction vector of vorticity  $\omega(x, t)/|\omega(x, t)|$ , which is defined in the region  $\{(x, t) \mid \omega(x, t) \neq 0\}$ . P. Constantin and C. Fefferman [9] first proved that if there exists a positive constant number  $C$  such that

$$\frac{\sqrt{1 - (\xi(x, t) \cdot \xi(x+h, t))^2}}{|h|} \leq C$$

holds in regions of high vorticity, then a solution to the Navier–Stokes equations is regular.

H. Beirão da Veiga, L. Berselli, and Y. Zhou [2, 3, 49, 50] improved this result.

*Assumption (A1).* There exist  $\beta \in [1/2, 1]$ , a positive constant  $K$ , and  $g \in L^a(0, T; L^b)$ , where

$$\frac{2}{a} + \frac{3}{b} = \beta - \frac{1}{2} \quad \text{with} \quad a \in \left[ \frac{4}{2\beta - 1}, \infty \right],$$

such that

$$\frac{\sqrt{1 - (\xi(x, t) \cdot \xi(x+h, t))^2}}{|h|^\beta} \leq g(x, t)$$

for  $(x, t), (x+h, t) \in \Omega_T(K) := \{(x, t) \in \mathbb{R}^3 \times (0, T) \mid |\omega(x, t)| > K\}$ .

*Assumption (A2).* There exist  $\beta \in (0, 1/2]$  and positive constants  $K$  and  $C$  such that

$$\frac{\sqrt{1 - (\xi(x, t) \cdot \xi(x+h, t))^2}}{|h|^\beta} \leq C$$

for  $(x, t), (x+h, t) \in \Omega_T(K)$ . Furthermore,

$$\omega \in L^q(0, T; L^r) \quad \text{for} \quad \frac{2}{q} + \frac{3}{r} \leq 2 + \beta, \quad \frac{3}{\beta + 2} \leq r < \frac{3}{\beta}.$$

**Theorem 3.1.2** ([2, 3, 49, 50]). Let  $v$  be a weak solution of the Navier–Stokes equations with a initial data  $v_0 \in H_\sigma^1$ , which means the Sobolev spaces of solenoidal vector fields. Suppose that the assumption (A1) or (A2) on the vorticity is satisfied. Then the solution is regular in  $(0, T)$ .

In [4], regularity theorems for various cases are mentioned. D. Chae [6] proved that if the direction of vorticity to  $(NS)_\alpha$  in the whole space is restricted by using some Triebel–Lizorkin norm  $\|\cdot\|_{\dot{F}_{b,p}^\beta}$  (see [6, pp.374]), then there is no singularity.

### 3. Refined Regularity Criterion

**Theorem 3.1.3** ([6]). Let  $\alpha \in (0, 2]$ . Let  $v$  be a strong solution of  $(\text{NS})_\alpha$  in the class  $E_s(T)$  with  $s > 5/2$ . For the  $v$ , let  $\omega$  be the vorticity and  $\xi$  be the direction vector of vorticity. Suppose there exists  $\beta \in (0, 1)$ ,  $p \in (3/(3 - \beta), \infty]$ ,  $b \in (1, \infty]$ ,  $r \in (1, 3/\beta)$  satisfying

$$\frac{\beta}{3} < \frac{1}{b} + \frac{1}{r} < \frac{\alpha + \beta}{3}, \quad \frac{1}{p} + \frac{1}{r} < 1 + \frac{\beta}{3}$$

and  $a, q \in [1, \infty]$  such that

$$\begin{aligned} \xi &\in L^a(0, T; \dot{\mathcal{F}}_{b,p}^\beta) \quad \text{and} \quad \omega \in L^q(0, T; L^r) \\ \text{with} \quad \frac{\alpha}{a} + \frac{\alpha}{q} + \frac{3}{b} + \frac{3}{r} &\leq \alpha + \beta. \end{aligned}$$

Then  $v$  can be continued to the strong solution in the class  $E_s(T')$  for some  $T < T'$ .

Here, M. Tanahashi et al. [42] pointed out the importance of fine scale structure in high vorticity regions on turbulence. Therefore, it is more important to obtain a continuation principle under some condition in regions of high vorticity.

Our result on continuation of strong solutions now reads;

**Theorem 3.1.4** (Main Theorem). Let  $s > 5/2$ ,  $T > 0$ ,  $\alpha \in (0, 2]$ . Let  $v$  be a strong solution of  $(\text{NS})_\alpha$  in the class  $E_s(T)$ . For the  $v$ , let  $\omega$  be the vorticity and  $\xi$  be the direction vector of vorticity. Let  $\beta \in (0, 1]$ ,  $a, b, q \in (1, \infty]$ ,  $r \in (1, 3/\beta)$  satisfy

$$\frac{1}{b} + \frac{1}{r} < \frac{\alpha + \beta}{3}, \quad \frac{\alpha}{a} + \frac{\alpha}{q} + \frac{3}{b} + \frac{3}{r} \leq \alpha + \beta.$$

Suppose that

(B1)  $\omega \in L^q(0, T; L^r)$ ,

(B2) there exist  $g \in L^a(0, T; L^b)$  and  $K > 0$  such that

$$\frac{\sqrt{1 - (\xi(x, t) \cdot \xi(x + h, t))^2}}{|h|^\beta} \leq g(x, t)$$

for  $(x, t), (x + h, t) \in \Omega_T(K) := \{(x, t) \in \mathbb{R}^3 \times (0, T) \mid |\omega(x, t)| > K\}$ .

Then  $v$  can be continued to the strong solution in the class  $E_s(T')$  for some  $T < T'$ .

Roughly speaking, Theorem 3.1.4 means that the assumption of regularity of the direction vector of vorticity in regions of high vorticity induces the regularity of velocity to  $(\text{NS})_\alpha$  as in the case of the usual Navier–Stokes equations.

*Remark 3.1.5.* Theorem 3.1.4 is the generalization of Theorem 3.1.2 for  $(\text{NS})_\alpha$ .

*Remark 3.1.6.* When  $\beta \in (0, 1)$ , the Triebel–Lizorkin norm  $\|\cdot\|_{\dot{\mathcal{F}}_{\infty, \infty}^\beta}$  is equivalent to the Hölder–Zygmund norm  $\|\cdot\|_{C^\beta}$  (see [44, Section 2.8], [6, Remark 1.4]). Theorem 3.1.4 with  $K = 0$  and  $a = b = \infty$  reduces to Theorem 3.1.3 with  $a = b = p = \infty$ .

For simplicity,  $a = b = \infty$ ,  $\beta = 5/2 - \alpha$ . It is known that  $(\text{NS})_2$  possess a global weak solution which belongs  $L^2(0, T; H^1)$  and a weak-strong uniqueness property was established [38]. Hence, when  $\alpha = 2$ , Theorem 3.1.4 reduces to Theorem 3.1.2. Furthermore, when  $\alpha = 5/2$ , it is obvious that the weak solution is automatically regular as J.L. Lions proved [28].

The proof of Theorem 3.1.4 is not a simple generalization of that of Theorem 3.1.2, since we cannot absorb the convection term into the viscosity term just by taking  $L^2$  inner product of (3.1.4) by  $\omega$  when  $\alpha < 3/2$ . To overcome this difficulty, we take  $L^2$  inner product of (3.1.4) by  $\omega|\omega|^{p-2}$ , where  $p \geq 3/\alpha (> 2)$ .

We assume the regularity of direction vorticity only for high vorticity regions. Therefore, we need to split the convection term based on  $K$  (see equation (3.3.4)). A  $J_2$ , which includes  $p$  high vorticity terms is a difficult term to estimate among the divided the convection terms, since we cannot use the assumption (B2) for estimating of the term  $J_2$ . Fortunately, in the usual Navier–Stokes equations case, we can use the energy estimate on  $\|\omega\|_2$  for this term  $J_2$  [2, 3, 49, 50]. However, we cannot obtain any available estimate on  $\|\omega\|_p$ . To overcome this difficulty, we use the assumption (B1) in order to estimate  $J_2$ .

### 3. Refined Regularity Criterion

#### 3.2 Deformation of the convection term

In this section we recall an important result proved by P. Constantin [8]. Let the strain matrix  $S$  be  $((\partial_{x_j} v_i + \partial_{x_i} v_j)/2)_{ij}$ . By using (3.1.5) we have

$$\begin{aligned} S &= \frac{3}{4\pi} P.V. \int_{\mathbb{R}^3} \frac{1}{2} (\hat{y} \otimes (\hat{y} \times \omega(x+y, t)) + (\hat{y} \times \omega(x+y, t)) \otimes \hat{y}) \frac{dy}{|y|^3} \\ &=: S_{[\omega]}(x, t). \end{aligned} \quad (3.2.1)$$

The integral is in the sense of principal value and  $\hat{y}$  is the direction vector of  $y$ . The tensor product  $\otimes$  denotes the matrix

$$(a \otimes b)_{ij} = a_i b_j \quad (a = (a_i)_i, b = (b_i)_i \in \mathbb{R}^3),$$

and  $\times$  is cross product. By the formula

$$(\omega \cdot \nabla) v \cdot \omega = S\omega \cdot \omega (=: J),$$

and (3.2.1), the convection term can be written by

$$\begin{aligned} &(\omega(x, t) \cdot \nabla) v(x, t) \cdot \omega(x, t) \\ &= \frac{3}{4\pi} P.V. \int_{\mathbb{R}^3} (\xi(x, t) \cdot \hat{y}) (\xi(x+y, t) \times \xi(x, t) \cdot \hat{y}) |\omega(x+y, t)| \frac{dy}{|y|^3} |\omega(x, t)|^2, \end{aligned} \quad (3.2.2)$$

where  $\xi := \omega/|\omega|$ . Based on this formula, we prove Theorem 3.1.4.

#### 3.3 Proof of Theorem 3.1.4

Let  $p > \max\{6/\alpha - 2, 3/\alpha, 2\}$ . Taking  $L^2(\mathbb{R}^3)$  inner product of (3.1.4) by  $\omega|\omega|^{p-2}$ , we have

$$\frac{1}{p} \partial_t \|\omega\|_p^p + \int_{\mathbb{R}^3} (-\Delta)^{\alpha/2} \omega \cdot \omega |\omega|^{p-2} dx = \int_{\mathbb{R}^3} (\omega \cdot \nabla) v \cdot \omega |\omega|^{p-2} dx. \quad (3.3.1)$$

We write  $J$  as the right hand side of (3.3.1). The viscosity term on the left hand side is estimated by

$$\begin{aligned} \int_{\mathbb{R}^3} |\omega|^{p-2} \omega \cdot (-\Delta)^{\alpha/2} \omega dx &\geq \frac{2}{p} \int_{\mathbb{R}^3} |(-\Delta)^{\alpha/4} |\omega|^{p/2}|^2 dx \\ &\geq C_1 \left( \int_{\mathbb{R}^3} |\omega|^{3p/(3-\alpha)} dx \right)^{(3-\alpha)/3} = C_1 \|\omega\|_{\frac{3p}{3-\alpha}}^p, \end{aligned} \quad (3.3.2)$$

where we used [24, Lemma 3.3] for the estimate of the fractional derivative in the first inequality, and the Sobolev embedding in the second inequality. We note that  $C_1$  depends on  $\alpha$  and  $p$ .

Let  $K$  be a positive constant in Theorem 3.1.4. We split  $\omega$  into  $\omega = \omega_{(1)} + \omega_{(2)}$ , where

$$\begin{aligned} \omega_{(1)}(x, \cdot) &= \begin{cases} \omega(x, \cdot), & \text{if } |\omega(x, \cdot)| \leq K \\ 0, & \text{if } |\omega(x, \cdot)| > K \end{cases}, \\ \omega_{(2)}(x, \cdot) &= \begin{cases} 0, & \text{if } |\omega(x, \cdot)| \leq K \\ \omega(x, \cdot), & \text{if } |\omega(x, \cdot)| > K \end{cases}. \end{aligned}$$

Let us decompose  $S_{[\omega]}(x, t) = S_{(1)}(x, t) + S_{(2)}(x, t)$ , where

$$\begin{aligned} S_{(1)}(x, t) &:= S_{[\omega_{(1)}]}(x, t), \\ S_{(2)}(x, t) &:= S_{[\omega_{(2)}]}(x, t). \end{aligned}$$

Note that, by the Calderón-Zygmund inequality [5],

$$\|S_{(i)}(t)\|_{\zeta} \leq C_2 \|\omega_{(i)}(t)\|_{\zeta} \quad (\zeta \in (1, \infty), i = 1, 2), \quad (3.3.3)$$

### 3. Refined Regularity Criterion

where  $C_2$  depends on  $\zeta$ . Let us decompose the convection term  $J$  into the following three parts.

$$\begin{aligned} J_1 &:= \int_{\mathbb{R}^3} S_{(2)} \xi \cdot \xi |\omega_{(2)}|^p dx, \\ J_2 &:= \int_{\mathbb{R}^3} S_{(1)} \xi \cdot \xi |\omega_{(2)}|^p dx, \\ J_3 &:= \sum_{i=1,2} \int_{\mathbb{R}^3} S_{(i)} \xi \cdot \xi |\omega_{(1)}|^p dx. \end{aligned} \quad (3.3.4)$$

The most difficult term is  $J_1$ . We need the help of the assumption (B2). From (3.2.2) we get

$$\begin{aligned} &S_{(2)}(x, t) \omega_{(2)}(x, t) \cdot \omega_{(2)}(x, t) \\ &= \frac{3}{4\pi} P.V. \int_{\mathbb{R}^3} \frac{1}{2} (\hat{y} \otimes (\hat{y} \times \omega_{(2)}(x+y, t)) \\ &\quad + (\hat{y} \times \omega_{(2)}(x+y, t)) \otimes \hat{y}) \omega_{(2)}(x, t) \cdot \omega_{(2)}(x, t) \frac{dy}{|y|^3} \\ &= \frac{3}{4\pi} P.V. \int_{\mathbb{R}^3} (\xi(x, t) \cdot \hat{y})(\xi(x+y, t) \times \xi(x, t) \cdot \hat{y}) \\ &\quad |\omega_{(2)}(x+y, t)| \frac{dy}{|y|^3} |\omega_{(2)}(x, t)|^2. \end{aligned}$$

Using the assumption (B2) and the Hölder inequality, we get

$$\begin{aligned} |J_1| &= \frac{3}{4\pi} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} |\xi(x+y, t) \times \xi(x, t)| |\omega_{(2)}(x+y, t)| \frac{dy}{|y|^3} |\omega_{(2)}(x, t)|^p dx \\ &\leq \frac{3}{4\pi} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} |g(x, t)| \frac{|\omega(x+y, t)|}{|y|^{3-\beta}} dy |\omega(x, t)|^p dx \\ &\leq \frac{3}{4\pi} \|g\|_b \|I_\beta(|\omega|)\|_\delta \|\omega\|_{pk}^p, \end{aligned}$$

where  $b, \delta$ , and  $k$  satisfy

$$\frac{1}{b} + \frac{1}{\delta} + \frac{1}{k} = 1, \quad b, \delta, k \geq 1, \quad (3.3.5)$$

and  $I_\beta(\cdot), 0 < \beta < 3$ , are the operator defined by the Riesz potential as follows.

$$I_\beta(|\omega|)(x) := \gamma(\beta) \int_{\mathbb{R}^3} \frac{|\omega(x+y)|}{|y|^{3-\beta}} dy, \quad \gamma(\beta) := 2^\beta \pi^{3/2} \frac{\Gamma(\frac{\beta}{2})}{\Gamma(\frac{3-\beta}{2})}.$$

Using the Hardy-Littlewood-Sobolev inequality [40, Chapter 5], we have

$$\|I_\beta(|\omega|)\|_\delta \leq C_3 \|\omega\|_r,$$

where

$$\frac{1}{\delta} = \frac{1}{r} - \frac{\beta}{3}, \quad \delta \in (1, \infty), \quad r \in \left(1, \frac{3}{\beta}\right). \quad (3.3.6)$$

On the other hand, using the standard  $L^p$ -interpolation inequality, we have

$$\|\omega\|_{pk}^p \leq \|\omega\|_p^{\frac{3+k(\alpha-3)}{\alpha k}} \|\omega\|_{\frac{3p}{3-\alpha}}^{\frac{3(k-1)}{\alpha k} p},$$

where  $k < \frac{3}{3-\alpha}$ . Furthermore, using the Young inequality and the relation (3.3.5) and (3.3.6), we have

$$\begin{aligned} |J_1| &\leq C_4 \|g\|_b^{\frac{\alpha k}{\alpha k - 3k + 3}} \|\omega\|_r^{\frac{\alpha k}{\alpha k - 3k + 3}} \|\omega\|_p^p + \frac{C_1}{4} \|\omega\|_{\frac{3p}{3-\alpha}}^p \\ &= C_4 \|g\|_b^{\frac{\alpha}{\alpha + \beta - \frac{3}{\beta} - \frac{3}{r}}} \|\omega\|_r^{\frac{\alpha}{\alpha + \beta - \frac{3}{\beta} - \frac{3}{r}}} \|\omega\|_p^p + \frac{C_1}{4} \|\omega\|_{\frac{3p}{3-\alpha}}^p, \end{aligned} \quad (3.3.7)$$

### 3. Refined Regularity Criterion

where  $C_4$  depends on  $\alpha$ ,  $\beta$ , and  $r$ .

We choose  $\mu \in (1, \min\{\frac{3}{3-\alpha}, \frac{r}{r-1}\})$  so that  $\frac{r(\mu-1)}{3-3\mu+\mu\alpha} < \frac{1}{\alpha+\beta-\frac{3}{b}-\frac{3}{r}}$ . We get

$$\begin{aligned} |J_2| &\leq \|S_{(1)}\|_{\frac{\mu}{\mu-1}} \|\omega_{(2)}\|_{p\mu}^p \\ &\leq C_5 \|\omega_{(1)}\|_{\frac{\mu}{\mu-1}} \|\omega_{(2)}\|_{p\mu}^p \end{aligned}$$

by the Hölder inequality. Moreover, using the  $L^p$ -interpolation inequality and the Young inequality, we obtain

$$|J_2| \leq C_6 \|\omega_{(1)}\|_{\frac{\alpha\mu}{3-3\mu+\mu\alpha}} \|\omega\|_p^p + \frac{C_1}{4} \|\omega\|_{\frac{3p}{3-\alpha}}^p, \quad (3.3.8)$$

where  $C_6$  depends on  $\alpha$  and  $\mu$ . Using  $|\omega_{(1)}| \leq K$ , we obtain

$$|J_2| \leq C_7 \|\omega_{(1)}\|_r^{\frac{\alpha r(\mu-1)}{3-3\mu+\mu\alpha}} \|\omega\|_p^p + \frac{C_1}{4} \|\omega\|_{\frac{3p}{3-\alpha}}^p, \quad (3.3.9)$$

where  $C_7$  depends on  $\alpha$ ,  $K$ ,  $\mu$ , and  $r$ .

A direct calculation yields

$$\begin{aligned} |J_3| &\leq \sum_{i=1,2} K \int_{\mathbb{R}^3} |S_{(i)}| |\omega_{(1)}|^{p-1} dx \\ &\leq \sum_{i=1,2} K \|S_{(i)}\|_p \|\omega_{(1)}\|_p^{p-1} \\ &\leq C_8 \|\omega\|_p^p, \end{aligned} \quad (3.3.10)$$

by using the Hölder inequality and (3.3.3) with  $\zeta = p$ . We note that  $C_8$  depends only on  $K$  and  $p$ .

Combining (3.3.1) with (3.3.2), (3.3.7), (3.3.9), and (3.3.10), we derive

$$\begin{aligned} \partial_t \|\omega\|_p^p + \frac{C_1}{2} \|\omega\|_{\frac{3p}{3-\alpha}}^p &\leq C_4 \|g\|_b^{\frac{\alpha}{\alpha+\beta-\frac{3}{b}-\frac{3}{r}}} \|\omega\|_r^{\frac{\alpha}{\alpha+\beta-\frac{3}{b}-\frac{3}{r}}} \|\omega\|_p^p \\ &\quad + C_7 \|\omega\|_r^{\frac{\alpha r(\mu-1)}{3-3\mu+\mu\alpha}} \|\omega\|_p^p + C_8 \|\omega\|_p^p. \end{aligned} \quad (3.3.11)$$

The Gronwall lemma applied to (3.3.11) with the Hölder inequality provides

$$\begin{aligned} &\|\omega(t)\|_p^p \\ &\leq \|\omega(0)\|_p^p \exp \left[ \int_0^T C_4 \|g(\tau)\|_b^{\frac{\alpha}{\alpha+\beta-\frac{3}{b}-\frac{3}{r}}} \|\omega(\tau)\|_r^{\frac{\alpha}{\alpha+\beta-\frac{3}{b}-\frac{3}{r}}} \right. \\ &\quad \left. + C_7 \|\omega(\tau)\|_r^{\frac{\alpha r(\mu-1)}{3-3\mu+\mu\alpha}} + C_8 d\tau \right] \\ &\leq \|\omega(0)\|_p^p \exp \left[ C_4 T^{C_9} \left\{ \int_0^T \|g(\tau)\|_b^{\frac{k_1 \alpha}{\alpha+\beta-\frac{3}{b}-\frac{3}{r}}} d\tau \right\}^{\frac{1}{k_1}} \right. \\ &\quad \times \left. \left\{ \int_0^T \|\omega(\tau)\|_r^{\frac{k_2 \alpha}{\alpha+\beta-\frac{3}{b}-\frac{3}{r}}} d\tau \right\}^{\frac{1}{k_2}} \right. \\ &\quad \left. + C_7 T^{C_{10}} \left\{ \int_0^T \|\omega(\tau)\|_r^{\frac{k_2 \alpha}{\alpha+\beta-\frac{3}{b}-\frac{3}{r}}} d\tau \right\}^{\frac{r(\mu-1)(\alpha+\beta-\frac{3}{b}-\frac{3}{r})}{(3-3\mu+\mu\alpha)k_2}} + C_8 T \right] \\ &< \infty, \end{aligned} \quad (3.3.12)$$

for all  $t \in [0, T]$  where  $1/k_1 + 1/k_2 \leq 1$  for  $k_1, k_2 > 1$ , by the assumptions (B1) and (B2). Note that  $C_9$  depends on  $\alpha$ ,  $\beta$ ,  $b$ ,  $k_1$ ,  $k_2$ , and  $r$  and that  $C_{10}$  depends on  $\alpha$ ,  $\beta$ ,  $b$ ,  $k_2$ ,  $\mu$ , and  $r$ . Finally, integrating

## Bibliography

(3.3.11) over  $[0, T]$ , we obtain

$$\begin{aligned} & \|\omega(T)\|_p^p + \frac{C_1}{2} \int_0^T \|\omega(\tau)\|_{\frac{3p}{3-\alpha}}^p d\tau \\ & \leq \|\omega(0)\|_p^p + \sup_{t \in [0, T]} \|\omega(t)\|_p^p \int_0^T C_4 \|g(\tau)\|_b^{\frac{\alpha}{\alpha+\beta-\frac{3}{b}-\frac{3}{r}}} \|\omega(\tau)\|_r^{\frac{\alpha}{\alpha+\beta-\frac{3}{b}-\frac{3}{r}}} \\ & \quad + C_7 \|\omega(\tau)\|_r^{\frac{\alpha r(\mu-1)}{3-3\mu+\mu\alpha}} + C_8 d\tau \\ & < \infty, \end{aligned}$$

by (3.3.12) and the assumptions (B1) and (B2). So, we get

$$\int_0^T \|\omega(t)\|_{\frac{3p}{3-\alpha}}^p dt < \infty.$$

Consequently, the vorticity satisfies (3.1.6). Hence, applying Theorem 3.1.1, we find that  $v$  can be continued to the strong solution in the class  $E_s(T')$  for some  $T < T'$ .

□

## Acknowledgment

The author is grateful to Professor Yong Zhou for telling me information. He also thanks Professor Tsuyoshi Yoneda for kind help for his research on the Navier–Stokes equations, and Professor Takahito Kashiwabara for useful comments. He was supported by the Leading Graduate Course for Frontiers of Mathematical Sciences and Physics (FMSP) at the University of Tokyo.

# Bibliography

- [1] Piotr Antonik, Marvyn Gulina, Jaël Pauwels, and Serge Massar. Using a reservoir computer to learn chaotic attractors, with applications to chaos synchronization and cryptography. *Phys. Rev. E*, 98:012215, Jul 2018.
- [2] Hugo Beirão da Veiga. Vorticity and smoothness in viscous flows. In *Nonlinear problems in mathematical physics and related topics, II*, volume 2 of *Int. Math. Ser. N. Y.*, pages 61–67. Kluwer/Plenum, New York, 2002.
- [3] Hugo Beirão da Veiga and Luigi C. Berselli. On the regularizing effect of the vorticity direction in incompressible viscous flows. *Differential Integral Equations*, 15(3):345–356, 2002.
- [4] Hugo Beirão da Veiga, Yoshikazu Giga, and Zoran Grujić. Vorticity direction and regularity of solutions to the navier-stokes equations. *Handbook of Mathematical Analysis in Mechanics of Viscous Fluids*, pages 901–932, 2018.
- [5] A. P. Calderón and A. Zygmund. On singular integrals. *Amer. J. Math.*, 78:289–309, 1956.
- [6] Dongho Chae. On the regularity conditions for the Navier-Stokes and related equations. *Rev. Mat. Iberoam.*, 23(1):371–384, 2007.
- [7] Alexandre Joel Chorin. The evolution of a turbulent vortex. *Comm. Math. Phys.*, 83(4):517–535, 1982.
- [8] Peter Constantin. Geometric statistics in turbulence. *SIAM Rev.*, 36(1):73–98, 1994.
- [9] Peter Constantin and Charles Fefferman. Direction of vorticity and the problem of global regularity for the Navier-Stokes equations. *Indiana Univ. Math. J.*, 42(3):775–789, 1993.
- [10] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.
- [11] Patricio Clark Di Leoni, Andrea Mazzino, and Luca Biferale. Inferring flow parameters and turbulent configuration with physics-informed data assimilation and spectral nudging. *Physical Review Fluids*, 3(10):104604, 2018.
- [12] Jishan Fan and Tohru Ozawa. On the regularity criteria for the generalized Navier-Stokes equations and Lagrangian averaged Euler equations. *Differential Integral Equations*, 21(5-6):443–457, 2008.
- [13] Uriel Frisch, Pierre Louis Sulem, and Mark Nelkin. A simple dynamical model of intermittent fully developed turbulence. *Journal of Fluid Mechanics*, 87(4):719–736, 1978.
- [14] Hunter Gabbard, Michael Williams, Fergus Hayes, and Chris Messenger. Matching matched filtering with deep networks for gravitational-wave astronomy. *Phys. Rev. Lett.*, 120(14):141103, 2018.
- [15] Masataka Gamahara and Yuji Hattori. Searching for turbulence models by artificial neural network. *Phys. Rev. Fluids*, 2(5):054604, 2017.
- [16] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

## Bibliography

- [17] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.
- [18] D Ibáñez-Soria, J Garcia-Ojalvo, A Soria-Frisch, and G Ruffini. Detection of generalized synchronization using echo state networks. *Chaos*, 28(3):033118, 2018.
- [19] Masanobu Inubushi and Kazuyuki Yoshimura. Reservoir computing beyond memory-nonlinearity trade-off. *Scientific Reports*, 7:10199, 2017.
- [20] Takashi Ishihara and Yukio Kaneda. High resolution dns of incompressible homogeneous forced turbulence—time dependence of the statistics—. In *Statistical Theories and Computational Approaches to Turbulence*, pages 177–188. Springer, 2003.
- [21] Keiichi Ishioka. ispack-0.4.1. <http://www.gfd-dennou.org/arch/ispack/>,, 1999. GFD Dennou Club.
- [22] Herbert Jaeger. The "echo state" approach to analysing and training recurrent neural networks. *GMD Report*, 148:13, 2001.
- [23] Herbert Jaeger and Harald Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 304:78–80, 2004.
- [24] Ning Ju. The maximum principle and the global attractor for the dissipative 2D quasi-geostrophic equations. *Comm. Math. Phys.*, 255(1):161–181, 2005.
- [25] N. H. Katz and N. Pavlović. A cheap Caffarelli-Kohn-Nirenberg inequality for the Navier-Stokes equation with hyper-dissipation. *Geom. Funct. Anal.*, 12(2):355–379, 2002.
- [26] J. N. Kutz. Deep learning in fluid dynamics. *J. Fluid Mech.*, 814:1–4, 2017.
- [27] J. Ling, A. Kurzawski, and J. Templeton. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *J. Fluid Mech.*, 807:155–166, 2016.
- [28] J. L. Lions. *Quelques méthodes de résolution des problèmes aux limites non linéaires*. Dunod; Gauthier-Villars, Paris, 1969.
- [29] Zhixin Lu, Brian R. Hunt, and Edward Ott. Attractor reconstruction by machine learning. *Chaos*, 28(6):061104, 2018.
- [30] Zhixin Lu, Jaideep Pathak, Brian Hunt, Michelle Girvan, Roger Brockett, and Edward Ott. Reservoir observers: Model-free inference of unmeasured variables in chaotic systems. *Chaos*, 27:041102, 2017.
- [31] M. Lukosevicius and Herbert Jaeger. Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3:127–149, 2009.
- [32] Wolfgang Maass, Thomas Natschläger, and Henry Markram. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 14:2531–2560, 2002.
- [33] Kyle Mills and Isaac Tamblyn. Deep neural networks for direct, featureless learning through observation: The case of two-dimensional spin models. *Phys. Rev. E*, 97(3):032119, 2018.
- [34] Kengo Nakai and Yoshitaka Saiki. Machine-learning inference of fluid variables from data using reservoir computing. *Phys. Rev. E*, 98:023111, Aug 2018.
- [35] Jaideep Pathak, Brian Hunt, Michelle Girvan, Zhixin Lu, and Edward Ott. Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Phys. Rev. Lett.*, 120:024102, 2018.
- [36] Jaideep Pathak, Zhixin Lu, Brian Hunt, Michelle Girvan, and Edward Ott. Using machine learning to replicate chaotic attractors and calculate lyapunov exponents from data. *Chaos*, 27:121102, 2017.
- [37] T Sauer, J A Yorke, and M Casdagli. Embedology. *J. Stat. Phys.*, 65:579–616, 1991.

## Bibliography

- [38] James Serrin. The initial value problem for the Navier-Stokes equations. In *Nonlinear Problems (Proc. Sympos., Madison, Wis., 1962)*, pages 69–98. Univ. of Wisconsin Press, Madison, Wis., 1963.
- [39] John C Snyder, Matthias Rupp, Katja Hansen, Klaus-Robert Müller, and Kieron Burke. Finding density functionals with machine learning. *Phys. Rev. Lett.*, 108(25):253002, 2012.
- [40] Elias M. Stein. *Singular integrals and differentiability properties of functions*. Princeton Mathematical Series, No. 30. Princeton University Press, Princeton, N.J., 1970.
- [41] F Takens. Detecting strange attractors in turbulence. In *Dynamical systems and turbulence, Warwick 1980 (Coventry, 1979/1980)*, volume 898 of *Lecture Notes in Math.*, pages 366–381. Springer, Berlin-New York, 1981.
- [42] M. Tanahashi, T. Miyauchi, and J. Ikeda. Scaling law of coherent fine scale structure in homogeneous isotropic turbulence. In *Proc. 11th symp. Turbulence Shear Flows (1997)*, 1997.
- [43] Andrey N. Tikhonov and Vasiliy Y. Arsenin. *Solutions of ill-posed problems*. V. H. Winston & Sons, Washington, D.C.: John Wiley & Sons, New York-Toronto, Ont.-London, 1977. Translated from the Russian, Preface by translation editor Fritz John, Scripta Series in Mathematics.
- [44] Hans Triebel. *The structure of functions*, volume 97 of *Monographs in Mathematics*. Birkhäuser Verlag, Basel, 2001.
- [45] D Verstraeten, B Schrauwen, M D’Haene, and D A Stroobandt. An experimental unification of reservoir computing methods. *Neural Network*, 20:391–403, 2007.
- [46] Pantelis R. Vlachas, Wonmin Byeon, Zhong Y. Wan, Themistoklis P. Sapsis, and Petros Koumoutsakos. Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 474(2213):20170844, 2018.
- [47] Z Y Wan, Pantelis Vlachas, Petros Koumoutsakos, and Themistoklis Sapsis. Data-assisted reduced-order modeling of extreme events in complex dynamical systems. *PloS one*, 13(5):e0197704, 2018.
- [48] Jiahong Wu. Generalized MHD equations. *J. Differential Equations*, 195(2):284–312, 2003.
- [49] Yong Zhou. Direction of vorticity and a new regularity criterion for the Navier-Stokes equations. *ANZIAM J.*, 46(3):309–316, 2005.
- [50] Yong Zhou. A new regularity criterion for the Navier-Stokes equations in terms of the direction of vorticity. *Monatsh. Math.*, 144(3):251–257, 2005.
- [51] Yong Zhou. Regularity criteria for the generalized viscous MHD equations. *Ann. Inst. H. Poincaré Anal. Non Linéaire*, 24(3):491–505, 2007.