Lyapunov exponents and Lyapunov vectors of a dynamical system constructed by machine learning

Yoshitaka Saiki (Hitotsubashi University)

1 Introduction.

Reservoir computing, a brain-inspired machine-learning technique that employs a data-driven dynamical system, is effective in predicting time series and frequency spectra in chaotic behaviors, including fluid flow and global atmospheric dynamics [2, 3, 5, 6].

The extent to which a data-driven model using reservoir computing can capture the dynamical properties of original systems should be determined. Lu *et al.* [2] reported that a data-driven model has an attractor similar to that of the original system under an appropriate choice of parameters. Nakai and Saiki [4] confirmed that a single data-driven model could infer the time series of chaotic fluid flow from various initial conditions. Zhu *et al.* [8] identified some unstable periodic orbits of a data-driven model through delayed feedback control. They suggested that a data-driven model could reconstruct the attractor of the original dynamical system.

This study clarifies that a data-driven model using reservoir computing has richer information than that obtained from a training data, especially from dynamical system point of view, suggesting that dynamical properties of the original unknown dynamical system can be estimated by reservoir computing from a relatively short time series. The dynamical properties, such as Lyapunov exponents and manifold structures between stable and unstable manifolds, can be reconstructed by the data-driven model through reservoir computing.

2 Reservoir computation.

What's Reservoir computation?

- a relatively high-dimentional fixed neural-network composed of simple nonlinear dynamical systems
- determination of output layer
- For Lorenz system and Kuramoto–Sivashinsky system, inference [2, 5, 6]

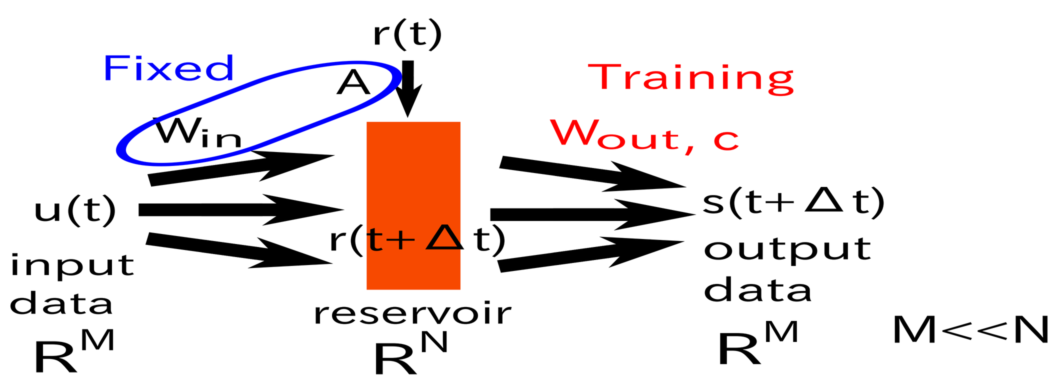
2.1 Procedure of training.

○ **1st step (making a reservoir vector)**

Making a reservoir vector $\mathbf{r}(t)$ correspond to decomposition of the input data $\mathbf{u}$ by using nonlinear function tanh:

$\mathbf{r}(t+\Delta t) = (1-\alpha)\mathbf{r}(t) + \alpha \tanh(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{\rm in}\mathbf{u}(t))$.

$\mathbf{A}, \mathbf{W}_{\rm in}$: sparse random matrix, whose maximal eigenvalue is controlled.



○ **2nd step (determination of output layer)**

We determine $\mathbf{W}_{\rm out}$ s.t.

$$t^{\forall} < T \qquad \mathbf{W}_{\rm out}\mathbf{r}(t+\Delta t) \approx \mathbf{s}(t+\Delta t).$$

⇒ Determine them s.t. the form is minimized:

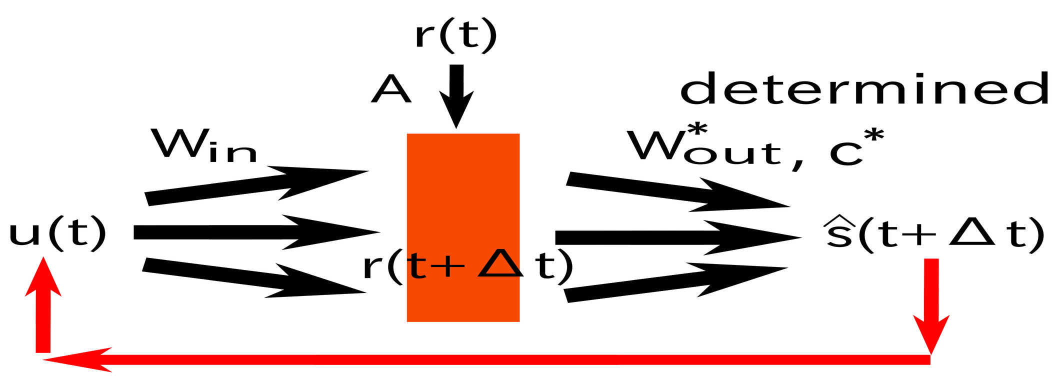
$$\sum_{l=1}^{L} \|(\mathbf{W}_{\rm out}\mathbf{r}(l\Delta t)) - \mathbf{s}(l\Delta t)\|^2 + \beta[Tr(\mathbf{W}_{\rm out}\mathbf{W}_{\rm out}^T)].$$

2.2 Procedure of inference.

Using the $\mathbf{W}^*_{\rm out}$, we infer the time-series $\mathbf{s}$.

$\hat{\mathbf{s}}(t) = \mathbf{W}^*_{\rm out}\mathbf{r}(t)$

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



This reservoir system corresponds to the data-driven model of u.

3 Lyapunov exponents and Lyapunov vectors.

The Lyapunov exponents are used to evaluate the degree of instability and estimate the Lyapunov dimension of a dynamical system. In some studies, the Lyapunov exponents of a data-driven model by reservoir computing were calculated in the space of N-dimensional reservoir state vector [1, 5, 6, 7]. Pathak *et al.* [5] computed Lyapunov exponents for the reservoir state vector and found that they almost coincide with those of the original system for the case of a partial differential equation, whereas only positive and neutral exponents coincide with those for the Lorenz system. To the best of the authors' knowledge, they have not been computed in a space of output variables.

In this study, an attempt is made to compute Lyapunov exponents in the space of output variables corresponding to x, y and z for the Lorenz system. Here we describe how to compute Lyapunov exponents and vectors in the original variables numerically from a trajectory of the data-driven model. We first estimate the Jacobian matrix at each point (x, y, z) along the trajectory of the data-driven model as follows:

(i) Apply the Taylor series expansion of order six to estimate $\dot{x} = dx/dt$, $\dot{y} = dy/dt$ and $\dot{z} = dz/dt$ at each sample point along the discrete trajectory;

(ii) Apply linear regression to the estimated values of $\dot{x}$, $\dot{y}$ and $\dot{z}$ by $x^l y^m z^n$ $(0 \le l+m+n \le 3,\ l, m, n \ge 0)$ as explanatory variables;

(iii) Obtain the Jacobian matrix $J(\mathbf{x})$ at each point $\mathbf{x}$ by differentiating polynomials with the regression coefficients estimated in (ii).

We compute Lyapunov exponents and vectors by integrating the linear ordinary differential equation having coefficients determined by the Jacobian matrices $(\dot{\mathbf{x}}(t) = J(\mathbf{x}(t))\mathbf{x}(t))$, while the orbit is given by the trajectory of the data-driven model. Note that in this computation the discrete time trajectory points of a data-driven model are considered samples of the continuous time trajectory. For the high-accuracy computation with a rather large time step Δt of the reservoir computing, we employ four-stage and fourth-order Runge-Kutta method with time step $2\Delta t$ from the points along an orbit trajectory.

Reference

[1] C. Gallicchio, A. Micheli, and L. Silvestri, Neurocomputing 298, 34 (2018).

[2] Z. Lu, B. R. Hunt, and E. Ott, Chaos 28, 061104 (2018).

[3] K. Nakai and Y. Saiki, Physical Review E 98, 023111 (2018).

[4] K. Nakai and Y. Saiki, Discrete Contin. Dyn. Syst. S 14, 3641079 (2021).

[5] J. Pathak, Z. Lu, B. Hunt, M. Girvan, and E. Ott, Chaos 27, 121102 (2017).

[6] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, Phys. Rev. Lett. 120, 024102 (2018).

[7] V. Pyragas and K. Pyragas, Phys. Lett. A 384, 126591 (2020).

[8] Q. Zhu, H. Ma, and W. Lin, Chaos 29, 093125 (2019).