The Modelling Theory of Dynamical System Deep Learning

Jianping Li[1,2]

1 Frontiers Science Center for Deep Ocean Multi-spheres and Earth System (DOMES)/Key Laboratory of Physical Oceanography/Academy of Future Ocean/College of Oceanic and Atmospheric Sciences/Center for Ocean Carbon Neutrality, Ocean University of China, Qingdao 266100, China

2. Laboratory for Ocean Dynamics and Climate, Qingdao Marine Science and Technology Center, Qingdao 266237, China

Email: ljp@ouc.edu.cn

Abstract

This article mainly introduces the framework of the modelling theory of dynamical system deep learning (DSDL). On the basis of attractor theory, we construct a series of DSDL models by establishing the nonlinear extension relationships between delay and non-delay attractors, between non-delay attractor and differential attractor, and time reversal mapping as well. These DSDL models mainly contain the conventional nonlinear prediction models, time-lag models, differential attractor mapping models (the conventional differential attractor mapping models and time-lag differential attractor mapping models), and time reversal models (the conventional time reversal models and time-lag time reversal models). In addition, we discuss key variables and differences among dynamical, statistical, machine learning (or artificial intelligence) and DSDL models.

Introduction

Since Lorenz (1963) discovered the phenomenon of chaos, how to predict nonlinear chaotic dynamical systems has become an important issue due to their extreme sensitivity to initial values. Usually, when the control equations of a nonlinear dynamical system are known, numerical solution is the primary choice for solving this problem. However, when the control equations of a nonlinear dynamical system are unknown, other methods need to be used. In the era of big data, machine learning (ML, or artificial intelligence) has become an important rather than the only option to solve the problem. Due to the inherent limitations of ML methods, utilizing the inherent property of attractors in nonlinear chaotic dynamical systems (Li, 1997; Li and Chou, 1997, 1998, 2003) allows us to attempt to establish prediction models by combining the delay embedding theorem of attractors with the observed data of the system.

For a compact and finite-dimensional manifold, Takens (1981) gave the delay embedding theorem in state space reconstruction for obtaining complete information about the states of dynamical system in the observed time-series through the delay mapping. Robinson (2005) extended the Taken’s embedding theorem to infinite-dimensional partial differential equations (PDEs). Yap et al. (2014) and Eftekhari et al. (2018) further extended the Taken’s embedding theorem to the case in noisy conditions.

Komalapriya et al. (2008, 2010) presented the inverse delay embedding theorem. Ma et al. (2014, 2018) used the inverse delay embedding theorem to make prediction for short-term high-dimensional time series. This article attempts to establish the modeling theory of DSDL and various kinds of DSDL models using the inverse delay embedding theorem based on the attractor theory.

Basic Notation

Let i, j, k, m, and n be integers, $j \le m$, $j \le n$, and let the row vector be

$$X_{i(j:m)} = (x_{ij}, x_{i(j+1)}, \dots, x_{im}), \tag{1.1}$$

$$A_{i,k(j:m)} = (a_{i,kj}, a_{i,k(j+1)}, \dots, a_{i,km}), \tag{1.2}$$

The column vectors are

$$X_{(j:m)i} = (x_{ji}, x_{(j+1)i}, \dots, x_{mi})^{\mathrm{T}}, \tag{1.3}$$

$$A_{i,(j:m)k} = (a_{i,jk}, a_{i,(j+1)k}, \dots, a_{i,mk})^{\mathrm{T}}, \tag{1.4}$$

Where the upper subscript T represents transposition. The forward evolution matrix (the evolution matrix for short) is denoted as

$$X_{m\times(j:n)} = X_{(1:m)\times(j:n)} = \begin{pmatrix} x_{1j} & x_{1(j+1)} & \cdots & x_{1n} \\ x_{2j} & x_{2(j+1)} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{mj} & x_{m(j+1)} & \cdots & x_{mn} \end{pmatrix}$$
$$= (X_{1(j:n)}, X_{2(j:n)}, \dots, X_{m(j:n)})^{\mathrm{T}}$$
$$= (X_{(1:m)j}, X_{(1:m)(j+1)}, \dots, X_{(1:m)n}). \tag{1.5}$$

When $j = 1$, the evolution matrix is denoted as

$$X_{m\times n} = X_{(1:m)\times(1:n)} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix}$$
$$= (X_{1(1:n)}, X_{2(1:n)}, \dots, X_{m(1:n)})^{\mathrm{T}}$$
$$= (X_{(1:m)1}, X_{(1:m)2}, \dots, X_{(1:m)n}). \tag{1.6}$$

The forward univariate delay matrix (the univariate delay matrix for short) is denoted as

$$X^{\mathrm{D}}_{i,(j:m)\times(j:n)} = \begin{pmatrix} x_{ij} & x_{i(j+1)} & \cdots & x_{in} \\ x_{i(j+1)} & x_{i(j+2)} & \cdots & x_{i(n+1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{im} & x_{i(m+1)} & \cdots & x_{i(m+n-j)} \end{pmatrix}$$
$$= (X_{i(j:n)}, X_{i(j+1:n+1)}, \dots, X_{i(m:m+n-j)})^{\mathrm{T}}$$
$$= (X^{\mathrm{T}}_{i(j:m)}, X^{\mathrm{T}}_{i(j+1:m+1)}, \dots, X^{\mathrm{T}}_{i(n:m+n-j)}) \tag{1.7}$$

When $j = 1$, the univariate delay matrix is denoted as

$$X^{\mathrm{D}}_{i,m\times n} = X^{\mathrm{D}}_{i,(1:m)\times(1:n)} = \begin{pmatrix} x_{i1} & x_{i2} & \cdots & x_{in} \\ x_{i2} & x_{i3} & \cdots & x_{i(n+1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{im} & x_{i(m+1)} & \cdots & x_{i(m+n-1)} \end{pmatrix}$$
$$= (X_{i(1:n)}, X_{i(2:n+1)}, \dots, X_{i(m:m+n-1)})^{\mathrm{T}}$$
$$= (X^{\mathrm{T}}_{i(1:m)}, X^{\mathrm{T}}_{i(2:m+1)}, \dots, X^{\mathrm{T}}_{i(n:m+n-1)}). \tag{1.8}$$

The coefficient matrix is denoted as

$$A_{i,m\times n} = A_{i,(1:m)\times(1:m)} = \begin{pmatrix} a_{i,11} & a_{i,12} & \cdots & a_{i,1m} \\ a_{i,21} & a_{i,22} & \cdots & a_{i,2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i,m1} & x_{i,m2} & \cdots & a_{i,mm} \end{pmatrix}$$
$$= (A_{i,1(1:m)}, A_{i,2(1:m)}, \dots, A_{i,m(1:m)})^{\mathrm{T}}$$
$$= (A_{i,(1:m)1}, A_{i,(1:m)2}, \dots, A_{i,(1:m)m}). \tag{1.9}$$

The general coefficient matrix is denoted as

$$A_{i,m\times(j:n)} = A_{i,(1:m)\times(j:n)} = \begin{pmatrix} a_{i,1j} & a_{i,1(j+1)} & \cdots & a_{i,1n} \\ a_{i,2j} & a_{i,2(j+1)} & \cdots & a_{i,2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i,mj} & x_{i,m(j+1)} & \cdots & a_{i,mn} \end{pmatrix}$$
$$= (A_{i,1(j:n)}, A_{i,2(j:n)}, \dots, A_{i,m(j:n)})^{\mathrm{T}}$$
$$= (A_{i,(1:m)j}, A_{i,(1:m)(j+1)}, \dots, A_{i,(1:m)n}). \tag{1.10}$$

Dynamical Evolution Relationship Between Non-delay and Delay Attractors of Dynamical Systems and Corresponding Model Construction

► *Mapping relationship between non-delay and delay attractors*

Let the non-delay attractor evolution matrix of a m-dimensional dynamical system, i.e., the evolution time series matrix, be

$$X(t) = \begin{pmatrix} x_1(t_1) & x_1(t_2) & \cdots & x_1(t_n) \\ x_2(t_1) & x_2(t_2) & \cdots & x_2(t_n) \\ \vdots & \vdots & \vdots & \vdots \\ x_m(t_1) & x_m(t_2) & \cdots & x_m(t_n) \end{pmatrix}, \tag{2.1}$$

If the time series are observation data, the evolution matrix of the above system is simplified

$$X_{m\times n} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{pmatrix}. \tag{2.2}$$

Let the delay attractor evolution matrix of a single variable X_i be

$$X^{\mathrm{D}}_{i,m\times(1:n)} = \begin{pmatrix} x_{i1} & x_{i2} & \cdots & x_{in} \\ x_{i2} & x_{i2} & \cdots & x_{i(n+1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{im} & x_{i(m+1)} & \cdots & x_{i(m+n+1)} \end{pmatrix}. \tag{2.3}$$

According to the Takens' Theorem (1981) and its extensions by Robinson (2005), Yap et al. (2014), and Eftekhari et al. (2018), we can establish the dynamic evolution relationship between non-delay attractor and delay attractor below

$$\Phi_N: X_{m\times n} \rightarrow X^{\mathrm{D}}_{i,m\times(1:n)}, \tag{2.4}$$

where Φ_N is the nonlinear mapping, i.e.

$$\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix} \xrightarrow{\Phi_N} \begin{pmatrix} x_{i1} & x_{i2} & \cdots & x_{in} \\ x_{i2} & x_{i3} & \cdots & x_{i(n+1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{im} & x_{i(m+1)} & \cdots & x_{i(m+n-1)} \end{pmatrix},$$

or

$$\begin{pmatrix} f_1(x_{11}) & f_1(x_{12}) & \cdots & f_1(x_{1n}) \\ f_2(x_{21}) & f_2(x_{22}) & \cdots & f_2(x_{2n}) \\ \vdots & \vdots & \vdots & \vdots \\ f_m(x_{m1}) & f_m(x_{m2}) & \cdots & f_m(x_{mn}) \end{pmatrix} \rightarrow \begin{pmatrix} x_{i1} & x_{i2} & \cdots & x_{in} \\ x_{i2} & x_{i3} & \cdots & x_{i(n+1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{im} & x_{i(m+1)} & \cdots & x_{i(m+n-1)} \end{pmatrix}, \tag{2.5}$$

where f_i $(i = 1, 2, ..., m)$ are the nonlinear functions.

► *Linear prediction model*

Firstly, we consider the simplest linear case to provide reference for establishing nonlinear models in practice. From Eq. (2.5), one has

$$A_{i,m\times m}: X_{m\times m} = X^{\mathrm{D}}_{i,m\times m}, \tag{2.6}$$

where $i = 1, 2, ..., m$,

$$A_{i,m\times m} = \begin{pmatrix} a_{i,11} & a_{i,12} & \cdots & a_{i,1m} \\ a_{i,21} & a_{i,22} & \cdots & a_{i,2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i,m1} & a_{i,m2} & \cdots & a_{i,mm} \end{pmatrix} = (A_{i,1(1:m)}, A_{i,2(1:m)}, \ldots, A_{i,m(1:m)})^{\mathrm{T}}, \tag{2.7}$$

$$\begin{aligned} X_{m\times m} &= (X_{1(1:m)}, X_{2(1:m)}, \ldots, X_{m(1:m)})^{\mathrm{T}} \\ &= (X_{(1:m)1}, X_{(1:m)2}, \ldots, X_{(1:m)m}), \end{aligned} \tag{2.8}$$

$$\begin{aligned} X^{\mathrm{D}}_{i,m\times m} &= (X_{i(1:m)}, X_{i(2:m+1)}, \ldots, X_{i(m:2m-1)})^{\mathrm{T}} \\ &= (X^{\mathrm{T}}_{i,(1:m)}, X^{\mathrm{T}}_{i,(2:m+1)}, \ldots, X^{\mathrm{T}}_{i,(m:2m-1)}). \end{aligned} \tag{2.9}$$

A linear prediction model can be constructed using Eq. (2.6). Let $n > m$,

$$X_{m\times n} = (X_{m\times m} \quad X^{\mathrm{P}}_{m\times(m+1:n)}), \tag{2.10}$$

$$X^{\mathrm{D}}_{i,m\times n} = (X^{\mathrm{D}}_{i,m\times m} \quad X^{\mathrm{DP}}_{i,m\times(m+1:n)}), \tag{2.11}$$

where

$$X_{m\times m} = \begin{pmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mm} \end{pmatrix}, \tag{2.12}$$

$$X^{\mathrm{P}}_{m\times(m+1:n)} = \begin{pmatrix} x_{1(m+1)} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m(m+1)} & \cdots & x_{mn} \end{pmatrix}, \tag{2.13}$$

$$X^{\mathrm{D}}_{i,m\times m} = \begin{pmatrix} x_{i1} & \cdots & x_{im} \\ \vdots & \ddots & \vdots \\ x_{im} & \cdots & x_{i(2m-1)} \end{pmatrix}, \tag{2.14}$$

$$X^{\mathrm{DP}}_{i,(m+1:2m)\times(m+1:n)} = \begin{pmatrix} x_{i(m+1)} & \cdots & x_{in} \\ \vdots & \ddots & \vdots \\ x_{i(2m)} & \cdots & x_{i(n+m-1)} \end{pmatrix}. \tag{2.15}$$

Therefore, the above evolutionary relationship is represented as follows:

$$A_{i,m\times m} X_{m\times m} = X^{\mathrm{D}}_{i,m\times m}, \tag{2.16}$$

$$A_{i,m\times m} X^{\mathrm{P}}_{m\times(m+1:n)} = X^{\mathrm{DP}}_{i,(m+1:2m)\times(m+1:n)}. \tag{2.17}$$

Eq. (2.16) yields the coefficient matrix $A_{i,m\times m}$. Since $X^{\mathrm{DP}}_{i,(m+1:2m)\times(m+1:n)}$ contains the variables to be predicted, it is called as the sample prediction matrix of . For clarity, let

$$\widehat{X}^{\mathrm{D}}_{i,(m+1:2m)\times(m+1:n)} = X^{\mathrm{DP}}_{i,(m+1:2m)\times(m+1:n)}. \tag{2.18}$$

i.e., Eq. (2.17) can be expressed by

$$A_{i,m\times m} X^{\mathrm{DP}}_{(m+1:2m)\times(m+1:n)} = \widehat{X}^{\mathrm{D}}_{i,(m+1:2m)\times(m+1:n)}.$$

It can be easily written by the commonly used form of the prediction model as follows:

$$\widehat{X}^{\mathrm{D}}_{i,(m+1:2m)\times(m+1:n)} = A_{i,m\times m} X^{\mathrm{DP}}_{(m+1:2m)\times(m+1:n)}. \tag{2.19}$$

► *Nonlinear prediction models*

In fact, Φ_N is a nonlinear mapping, and in order to establish the dynamic evolution relationship of Eq. (2.5), some new variables need to be introduced. In practice, in order to obtain an explicit relationship, f_i can be set as elementary functions, usually power functions, sine and cosine functions or their combinations. We can construct a hierarchical structure based on the order of the monomial (denoted as L), where the sine or cosine function can be treated as an alpha-variable. For example, for the first layer $L = 1$, it introduces all first-order monomials (with a number of C^1_m), and the set composed of these first-order monomials is denoted as $\mathbb{L}_1$; For the second layer $L = 2$, on the basis of the first layer, all second-order monomials (with a number of C^2_{m+1}) are introduced, and the set composed of these second-order monomials is denoted as $\mathbb{L}_2$; For the -th layer, on the basis of the layers $\mathbb{L}_1 \cup \mathbb{L}_2 \cup \mathbb{L}_{L-1}$, all L-order monomials (with a number of C^{L}_{m+L-1}) are introduced, and the set composed of these L-order monomials is referred to as $\mathbb{L}_L$. If the sum of the numbers of all monomials in the nonlinear part of f_i is l, and its maximum number is $l=\sum_{i=1}^{L}C^{i}_{m+i-1}$, then l variables $x_i(i=m+1, ..., m+l)$ are introduced to correspond to these terms, and the following relationship can be established

$$A_{i,M\times M}X_{M\times M}=X^{\mathrm{D}}_{i,M\times M}, \tag{2.20}$$

where $M=m+l$, $i=1, ..., M$,

$$A_{i,M\times M}=(A_{i,1(1:M)}, A_{i,2(1:M)}, \ldots, A_{i,M(1:M)})^{\mathrm{T}}, \tag{2.21}$$

$$\begin{aligned}X_{M\times M}&=(X_{1(1:M)}, X_{2(1:M)}, \ldots, X_{M(1:M)})^{\mathrm{T}}\\&=(X_{(1:M)1}, X_{(1:M)2}, \ldots, X_{(1:M)M}),\end{aligned} \tag{2.22}$$

$$\begin{aligned}X^{\mathrm{D}}_{i,M\times M}&=(X_{i(1:M)}, X_{i(2:M+1)}, \ldots, X_{i(M:2M-1)})^{\mathrm{T}}\\&=(X^{\mathrm{T}}_{i(1:M)}, X^{\mathrm{T}}_{i(2:M+1)}, \ldots, X^{\mathrm{T}}_{i(M:2M-1)}).\end{aligned} \tag{2.23}$$

Then a prediction model can be constructed using Eq. (2.20). Let $N>M$,

$$X_{M\times N}=(X_{M\times M}\quad X^{\mathrm{P}}_{M\times(M+1:N)}), \tag{2.24}$$

$$X^{\mathrm{D}}_{i,M\times N}=(X^{\mathrm{D}}_{i,M\times M}\quad X^{\mathrm{DP}}_{i,M\times(M+1:N)}), \tag{2.25}$$

where

$$X_{M\times M}=\begin{pmatrix}x_{11} & \cdots & x_{1M}\\ \vdots & \ddots & \vdots\\ x_{M1} & \cdots & x_{MM}\end{pmatrix}, \tag{2.26}$$

$$X^{\mathrm{P}}_{M\times(M+1:N)}=\begin{pmatrix}x_{1(M+1)} & \cdots & x_{1N}\\ \vdots & \ddots & \vdots\\ x_{M(M+1)} & \cdots & x_{MN}\end{pmatrix}, \tag{2.27}$$

$$X^{\mathrm{D}}_{i,M\times M}=\begin{pmatrix}x_{i1} & \cdots & x_{iM}\\ \vdots & \ddots & \vdots\\ x_{iM} & \cdots & x_{i(2M-1)}\end{pmatrix}, \tag{2.28}$$

$$X^{\mathrm{DP}}_{i,(M+1:2M)\times(M+1:N)}=\begin{pmatrix}x_{i(M+1)} & \cdots & x_{iN}\\ \vdots & \ddots & \vdots\\ x_{i(2M)} & \cdots & x_{i(N+2M-1)}\end{pmatrix}. \tag{2.29}$$

Therefore, the above evolutionary relationship is expressed as follows:

$$A_{i,M\times M}X_{M\times M}=X_{i,M\times M}, \tag{2.30}$$

$$A_{i,M\times M}X^{\mathrm{P}}_{(M+1:2M)\times(M+1:N)}=X^{\mathrm{DP}}_{i,(M+1:2M)\times(M+1:N)}. \tag{2.31}$$

Here $X^{\mathrm{DP}}_{i,(M+1:2M)\times(M+1:N)}$ is the prediction matrix for the sample x_i. For simplicity and clarity, let

$$\hat{X}^{\mathrm{D}}_{i,(M+1:2M)\times(M+1:N)}=X^{\mathrm{DP}}_{i,(M+1:2M)\times(M+1:N)}. \tag{2.32}$$

Then Eq. (2.31) is expressed as

$$A_{i,M\times M}X^{\mathrm{P}}_{M\times(M+1:N)}=\hat{X}^{\mathrm{D}}_{i,(M+1:2M)\times(M+1:N)}. \tag{2.33}$$

The first row of Eq. (2.20) cannot be used for prediction, so a prediction model is established from the j-th ($j=2, ..., M$) row, that is, the following mapping is constructed:

$$\Phi_N: X_{M\times N}\to X^{\mathrm{D}}_{i,(j:M+j-1)\times(1:N)}, (j=2, \ldots, M). \tag{2.34}$$

For $j=2$,

$$\begin{aligned}&A_{i,2(1:M)}X_{M\times M}=X_{i(2:M+1)}\to A_{i,2(1:M)}X_{(1:M)(M+1)}=\hat{x}_{i(M+2)},\\&\qquad\to A_{i,2(1:M)}\hat{X}_{(1:M)(M+2)}=\hat{x}_{i(M+3)}\\&\qquad\to A_{i,2(1:M)}\hat{X}_{(1:M)(M+k)}=\hat{x}_{i(M+k+1)}, (k=2, 3, \ldots).\end{aligned} \tag{2.35}$$

This achieves the prediction of the variable $x_i(i=1, 2, ..., m)$. In fact, we can establish the following delay prediction model (referred to as the D_1 model, which means making predictions one step in advance):

$$\begin{cases}A_{i,2(1:M)}X_{M\times M}=X_{i(2:M+1)},\\ \hat{X}_{i(M+2)}=A_{i,2(1:M)}X_{i(M+1)},\\ \hat{X}_{i(M+k+1)}=A_{i,2(1:M)}\hat{X}_{i(M+k)}, (k=2, 3, \ldots).\end{cases} \tag{2.36}$$

where $i=1, 2, ..., M$. Although the M equations are established here, in fact, only the original m variables of the system need to be solved, and other nonlinear terms can be obtained immediately. We can also write Eq. (2.36) in a universal matrix form below:

$$\begin{cases}A_{(1:M),2(1:M)}X_{M\times M}=X_{(1:M)\times(2:M+1)},\\ \hat{X}_{(1:M)(M+2)}=A_{(1:M),2(1:M)}X_{(1:M)(M+1)},\\ \hat{X}_{(1:M)(M+k+1)}=A_{(1:M),2(1:M)}\hat{X}_{(1:M)(M+k)}, (k=2, 3, \ldots).\end{cases} \tag{2.37}$$

where

$$A_{(1:M),2(1:M)}=\begin{pmatrix}a_{1,21} & a_{1,22} & \cdots & a_{1,2M}\\ a_{2,21} & a_{2,22} & \cdots & a_{2,2M}\\ \vdots & \vdots & \vdots & \vdots\\ a_{M,21} & a_{M,22} & \cdots & a_{M,2M}\end{pmatrix},$$

$$X_{M\times M}=\begin{pmatrix}x_{11} & x_{12} & \cdots & x_{1M}\\ x_{21} & x_{22} & \cdots & x_{2M}\\ \vdots & \vdots & \vdots & \vdots\\ x_{M1} & x_{M2} & \cdots & x_{MM}\end{pmatrix},$$

$$X_{(1:M)\times(2:M+1)}=\begin{pmatrix}x_{12} & x_{13} & \cdots & x_{1(M+1)}\\ x_{22} & x_{23} & \cdots & x_{2(M+1)}\\ \vdots & \vdots & \vdots & \vdots\\ x_{M2} & x_{M3} & \cdots & x_{M(M+1)}\end{pmatrix}.$$

When $j>2$, the corresponding delay prediction models can also be established, e.g., for $j=3$

$$A_{i,3(1:M)}X_{M\times M} = X^{T}_{i(2:M+2)} \rightarrow A_{i,3(1:M)}X_{(1:M)(M+1)} = \hat{x}_{i(M+3)}$$
$$\rightarrow A_{i,3(1:M)}\hat{X}_{(1:M)(M+2)} = \hat{x}_{i(M+4)}$$
$$\rightarrow A_{i,3(1:M)}\hat{X}_{(1:M)(M+k)} = \hat{x}_{i(M+k+2)}, (k=2,3,\ldots). \quad (2.38)$$

For the general j (> 2), we have

$$A_{i,j(1:M)}X_{M\times M} = X^{T}_{i(j:M+j-1)} \rightarrow A_{i,j(1:M)}X_{(1:M)(M+1)} = \hat{x}_{i(M+j)}$$
$$\rightarrow A_{i,j(1:M)}\hat{X}_{(1:M)(M+2)} = \hat{x}_{i(M+j+1)}$$
$$\rightarrow A_{i,j(1:M)}\hat{X}_{(1:M)(M+k+j-2)} = \hat{x}_{i(M+k+j-1)}, (k=2,3,\ldots). \quad (2.39)$$

Therefore, when , the following delay prediction model (referred to as the model), which can make a prediction steps ahead, can be established below:

$$\begin{cases} A_{i,j(1:M)}X_{M\times M} = X_{i(j:M+j-1)}, \\ \hat{X}_{i(M+j)} = A_{i,j(1:M)}X_{i(M+1)}, \\ \hat{X}_{i(M+j+k-1)} = A_{i,j(1:M)}\hat{X}_{i(M+k)}, (k=2,3,\ldots). \end{cases} \quad (2.40)$$

where $i = 1, 2, ..., M$. We can also write Eq. (2.40) in a general matrix form as follows:

$$\begin{cases} A_{(1:M),j(1:M)}X_{M\times M} = X_{(1:M)\times(j:M+j-1)}, \\ \hat{X}_{(1:M)(M+j)} = A_{(1:M),j(1:M)}X_{(1:M)(M+1)}, \\ \hat{X}_{(1:M)(M+j+k+1)} = A_{(1:M),j(1:M)}\hat{X}_{(1:M)(M+k)}, (k=2,3,\ldots). \end{cases} \quad (2.41)$$

where

$$A_{(1:M),j(1:M)} = \begin{pmatrix} a_{1,j1} & a_{1,j2} & \cdots & a_{1,jM} \\ a_{2,j1} & a_{2,j2} & \cdots & a_{2,jM} \\ \vdots & \vdots & \vdots & \vdots \\ a_{M,j1} & a_{M,j2} & \cdots & a_{M,jM} \end{pmatrix},$$

$$A_{(1:M),j(j:M+j-1)} = \begin{pmatrix} x_{1j} & x_{1(j+1)} & \cdots & x_{1(M+j-1)} \\ x_{2j} & x_{2(j+1)} & \cdots & x_{2(M+j-1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{Mj} & x_{M(j+1)} & \cdots & x_{M(M+j-1)} \end{pmatrix}.$$

As can be seen from the above, when $j = 2$, the prediction model D_1 (i.e. Eq. (2.36) or Eq. (2.37)) uses the least number of observations. But when $j > 2$, the prediction model D_K (i.e. Eq. (2.40) or Eq. (2.41)) can predict $K = j - 1$ steps in advance.

For the D_1 model (i.e. Eq. (2.36)), the following predictor-corrector delay model (referred to as the PCD_1 model) can be established for predicting one step in advance:

$$\begin{cases} A_{i,2(1:M)}X_{M\times M} = X_{i(2:M+1)}, \\ \bar{X}_{i(M+2)} = A_{i,2(1:M)}X_{i(M+1)}, \\ \hat{X}_{i(M+2)} = \frac{1}{2}A_{i,2(1:M)}(X_{i(M+1)} + \bar{X}_{i(M+2)}), \\ \bar{X}_{i(M+k+1)} = A_{i,2(1:M)}\hat{X}_{i(M+k)}, \\ \hat{X}_{i(M+k+1)} = \frac{1}{2}A_{i,2(1:M)}(\hat{X}_{i(M+k)} + \bar{X}_{i(M+k+1)}), \\ (k=2,3,\ldots). \end{cases} \quad (2.42)$$

where $i = 1, 2, ..., M$. If it is written in general matrix form, Eq. (2.42) becomes:

$$\begin{cases} A_{(1:M),2(1:M)}X_{M\times M} = X_{(1:M)\times(2:M+1)}, \\ \bar{X}_{(1:M)(M+2)} = A_{(1:M),2(1:M)}X_{(1:M)(M+1)}, \\ \hat{X}_{(1:M)(M+2)} = \frac{1}{2}A_{(1:M),2(1:M)}(X_{(1:M)(M+1)} + \bar{X}_{(1:M)(M+2)}), \\ \bar{X}_{(1:M)(M+k+1)} = A_{(1:M),2(1:M)}\hat{X}_{(1:M)(M+k)}, \\ \hat{X}_{(1:M)(M+k+1)} = \frac{1}{2}A_{(1:M),2(1:M)}(\hat{X}_{(1:M)(M+k)} + \bar{X}_{(1:M)(M+k+1)}), \\ (k=2,3,\ldots). \end{cases} \quad (2.43)$$

Following the above, a predictor-corrector delay model PCD_K, which can predict $K = j - 1 (j > 2)$ steps in advance, can be established.

► *Key variables*

In the course of modeling, although there are many initial variables, in fact, many variables are irrelevant variables that need to be eliminated, and only the variables that are crucial to the system's evolutionary behavior are retained, which are called the key variables. How to find these key variables and remove irrelevant variables is the key to modeling and interpretability of the model, which requires the use of big data and the design of suitable judgment indicators for removing irrelevant variables. Assuming there is a suitable criterion system I for removing irrelevant variables, there are two ways to find the set of key variables (Fig. 1). One way is to research and judge the key variables in each layer (Fig. 1a); Another approach is to involve the set of key variables obtained from the previous layer in the search and determination of key variables in the next layer (Fig. 1b). The latter method may save computational resources when there are numerous system variable parameters. As for which method is more effective, it needs to be judged specifically in practice.

Time-lag Models

Essentially, the mapping (2.5) does not fully consider time-lag relationship, that is, does not fully utilize the previous information of the system, which may lead to a decrease in the accuracy of the model prediction. Therefore, it is necessary to extend the prediction model. Let $J \geq 0$, we construct the following time-lag mapping between previous

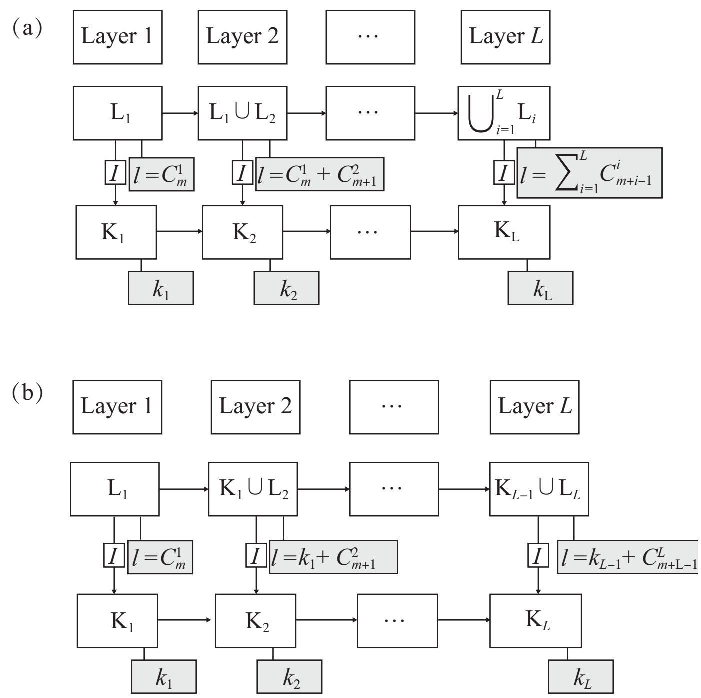

Fig. 1　Two ways to implement a set of key variables. (a) Redefine the set of key variables for each layer. (b) The set of key variables from the previous layer is involved in determining the key variables for the next layer. In the figure, $\mathbb{L}_i$ represents the set of i-th order monomials in the i-th layer, l represents the sum of all monomials in the corresponding layer, I is a determined indicator system for removing irrelevant variables, $\mathbb{K}_i$ is the set of key variables in the i-th layer, and k_i is the number of corresponding key variable

information evolution matrix and univariate delay matrix:

$$\Phi_N: X_{([-J:1]m)\times n} \rightarrow X^{\mathrm{D}}_{i,m\times(1:(J+2)n)}. \tag{3.1}$$

That is to say, we need to consider the information from the previous $2 + j$ steps. As above, let f_i still be elementary functions, so that Eq. (2.20) can be transformed into

$$A_{i,M\times([-J:1]M)} X_{([-J:1]M)\times(1:(J+2)M)} = X^{\mathrm{D}}_{i,M\times(1:(J+2)M)}. \tag{3.2}$$

where the coefficient matrix

$$A_{i,M\times([-J:1]M)} = (A_{i,M\times(1:M)}, A_{i,M\times(0:M-1)}, \ldots, A_{i,M\times(-J:M-J-1)}), \tag{3.3}$$

the previous information evolution matrix

$$X_{([-J:1]M)\times(1:(J+2)M)} = (X_{M\times(1:(J+2)M)}, X_{M\times(0:(J+2)M-1)}, \ldots, X_{M\times(-J:(J+2)M-J-1)})^{\mathrm{T}} = \begin{pmatrix} X_{M\times(1:(J+2)M)} \\ X_{M\times(0:(J+2)M-1)} \\ \vdots \\ X_{M\times(-J:(J+2)M-J-1)} \end{pmatrix}, \tag{3.4}$$

the univariate delay matrix

$$X^{\mathrm{D}}_{i,M\times(1:(J+2)M)} = (X_{i(1:(J+2)M)}, X_{i(2:(J+2)M+1)}, \ldots, X_{i(M:(J+2)M+M-1)}). \tag{3.5}$$

Similarly, the first row cannot be used for prediction. A corresponding prediction model can be established from the j-th (j = 2, ..., M) row. When j = 2, a time-lag prediction model (referred to as $\mathrm{TLD}_1(j + 2)$ model), which uses the information from the previous $2 + J$ steps to predict one step in advance, can be established as follows

$$\begin{cases} A_{i,2([-J:1]M)} X_{([-J:1]M)\times(1:(J+2)M)} = X_{i(2:(J+2)M+1)}, \\ \widehat{X}_{i((J+2)M+2)} = A_{i,2([-J:1]M)} X_{([-J:1]M)\times((J+2)M+1)}, \\ \widehat{X}_{i((J+2)M+k+1)} = A_{i,2([-J:1]M)} \widehat{X}_{([-J:1]M)((J+2)M+k)}, \\ \qquad (k = 2, 3, \ldots). \end{cases} \tag{3.6}$$

where i = 1, 2, ..., M,

$$A_{i,2([-J:1]M)} = (A_{i,2(1:M)}, A_{i,2(0:M-1)}, \ldots, A_{i,2(-J:M-J-1)}),$$

$$X_{i,(2:(J+2)M+1)} = (x_{i2}, x_{i3}, \ldots, x_{i(J+2)M+1}).$$

If it is written in general matrix form, Eq. (3.6) becomes:

$$\begin{cases} A_{(1:M),2([-J:1]M)}X_{([-J:1]M)\times(1:(J+2)M)} = X_{(1:M)\times(2:(J+2)M+1)}, \\ \hat{X}_{(1:M)((J+2)M+2)} = A_{(1:M),2([-J:1]M)}X_{([-J:1]M)\times((J+2)M+1)}, \\ \hat{X}_{(1:M)((J+2)M+k+1)} = A_{(1:M),2([-J:1]M)}\hat{X}_{([-J:1]M)((J+2)M+k)}, \\ \qquad (k = 2, 3, \ldots). \end{cases} \tag{3.7}$$

where

$$A_{(1:M),2([-J:1]M)} = \begin{pmatrix} A_{1,2(1:M)} & A_{1,2(0:M-1)} & \cdots & A_{1,2(-J:M-J-1)} \\ A_{2,2(1:M)} & A_{2,2(0:M-1)} & \cdots & A_{2,2(-J:M-J-1)} \\ \vdots & \vdots & \vdots & \vdots \\ A_{M,2(1:M)} & A_{M,2(0:M-1)} & \cdots & a_{M,2(-J:M-J-1)} \end{pmatrix}, \tag{3.8}$$

$$X_{(1:M)\times(2:(J+2)M+1)} = \begin{pmatrix} x_{12} & x_{13} & \cdots & x_{1((J+2)M+1)} \\ x_{22} & x_{23} & \cdots & x_{2((J+2)M+1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{M2} & x_{M3} & \cdots & x_{M((J+2)M+1)} \end{pmatrix}. \tag{3.9}$$

When $j > 2$, we establish a following time-lag delay prediction model (referred to as the $TLD_K(J + 2)$ model) that utilizes the information from the previous $2 + J$ steps to predict $K = j - 1$ steps in advance:

$$\begin{cases} A_{i,j([-J:1]M)}X_{([-J:1]M)\times(1:(J+2)M)} = X_{i(j:(J+2)M+j-1)}, \\ \hat{X}_{i((J+2)M+j)} = A_{i,j([-J:1]M)}X_{([-J:1]M)\times((J+2)M+1)}, \\ \hat{X}_{((-J:1)M)((J+2)M+j+k-1)} = A_{i,j([-J:1]M)}\hat{X}_{([-J:1]M)((J+2)M+k)}, \\ \qquad (k = 2, 3, \ldots). \end{cases} \tag{3.10}$$

where $i = 1, 2, \ldots, M$,

$$A_{i,j([-J:1]M)} = (A_{i,j(1:M)}, A_{i,j(0:M-1)}, \ldots, A_{i,j(-J:M-J-1)}),$$

$$X_{i,(j:(J+2)M+j-1)} = (x_{ij}, x_{i(j+1)}, \ldots, x_{i((J+2)M+j-1)}).$$

If it is written in a general matrix form, Eq. (3.10) becomes:

$$\begin{cases} A_{(1:M),j([-J:1]M)}X_{([-J:1]M)\times(1:(J+2)M)} = X_{(1:M)\times(j:(J+2)M+j-1)}, \\ \hat{X}_{(1:M)((J+2)M+j)} = A_{(1:M),j([-J:1]M)}X_{([-J:1]M)((J+2)M+1)}, \\ \hat{X}_{((-J:1)M)((J+2)M+j+k-1)} = A_{(1:M),j([-J:1]M)}\hat{X}_{([-J:1]M)((J+2)M+k)}, \\ \qquad (k = 2, 3, \ldots). \end{cases} \tag{3.11}$$

where

$$A_{(1:M),j([-J:1]M)} = \begin{pmatrix} A_{1,j(1:M)} & A_{1,j(0:M-1)} & \cdots & A_{1,j(-J:M-J-1)} \\ A_{2,j(1:M)} & A_{2,j(0:M-1)} & \cdots & A_{2,j(-J:M-J-1)} \\ \vdots & \vdots & \vdots & \vdots \\ A_{M,j(1:M)} & A_{M,j(0:M-1)} & \cdots & A_{M,j(-J:M-J-1)} \end{pmatrix}, \tag{3.12}$$

$$X_{(1:M)\times(j:(J+2)M+j-1)} = \begin{pmatrix} x_{1j} & x_{1(j+1)} & \cdots & x_{1((J+2)M+j-1)} \\ x_{2j} & x_{2(j+1)} & \cdots & x_{2((J+2)M+j-1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{Mj} & x_{M(j+1)} & \cdots & x_{M((J+2)M+j-1)} \end{pmatrix}. \tag{3.13}$$

For the TLD_1 model (i.e. Eq. (3.6)), the following predictor-corrector time-lag delay model (referred to as the $PCTLD_1(J + 2)$ model), which uses the information from the previous $J + 2$ steps to predict one step ahead, can be established below:

$$\begin{cases} A_{i,2([-J:1]M)}X_{([-J:1]M)\times(1:(J+2)M)} = X_{i(2:(J+2)M+1)}, \\ \overline{X}_{i((J+2)M+2)} = A_{i,2([-J:1]M)}X_{([-J:1]M)((J+2)M+1)}, \\ \hat{X}_{i((J+2)M+2)} = \frac{1}{2}A_{i,2([-J:1]M)}(X_{([-J:1]M)((J+2)M+1)} + \overline{X}_{i((J+2)M+2)}), \\ \overline{X}_{i((J+2)M+k+1)} = A_{i,2([-J:1]M)}\hat{X}_{([-J:1]M)((J+2)M+k)}, \\ \hat{X}_{i((J+2)M+k+1)} = \frac{1}{2}A_{i,2([-J:1]M)}(\hat{X}_{([-J:1]M)((J+2)M+k)}, \\ \quad + \overline{X}_{i(1:M)((J+2)M+k+1)}) \ (k = 2, 3, \ldots). \end{cases} \tag{3.14}$$

where $i = 1, 2, \ldots, M$. If it is written in a general matrix form, Eq. (3.14) becomes:

$$\begin{cases} A_{(1:M),2([-J:1]M)}X_{([-J:1]M)\times(1:(J+2)M)} = X_{(1:M)\times(2:(J+2)M+1)}, \\ \overline{X}_{(1:M)((J+2)M+2)} = A_{(1:M),2([-J:1]M)}X_{([-J:1]M)((J+2)M+1)}, \\ \hat{X}_{(1:M)((J+2)M+2)} = \frac{1}{2}A_{(1:M),2([-J:1]M)}(X_{([-J:1]M)((J+2)M+1)} \\ \quad + \overline{X}_{(1:M)((J+2)M+2)}), \\ \overline{X}_{(1:M)((J+2)M+k+1)} = A_{(1:M),2([-J:1]M)}\hat{X}_{([-J:1]M)((J+2)M+k)}, \\ \hat{X}_{(1:M)((J+2)M+k+1)} = \frac{1}{2}A_{(1:M),2([-J:1]M)}(\hat{X}_{([-J:1]M)((J+2)M+k)}, \\ \quad + \overline{X}_{(1:M)((J+2)M+k+1)}) \ (k = 2, 3, \ldots). \end{cases} \tag{3.15}$$

Following the above, a predictor-corrector time-lag delay model $PCTLD_K(J + 2)$, which uses the information from the previous $J + 2$ steps to predict $K = j - 1 (j > 2)$ steps ahead, can be established.

Differential Attractor Mapping Models

► *Conventional differential attractor mapping models*

In fact, there is a sampling time interval in the observation data, however, the mapping (2.5) does not consider this issue. The description of system evolution may vary depending on the sampling time interval, therefore, the sampling time interval should be considered. Let h be the sampling interval, we construct the following differential attractor mapping:

$$\Phi_N : hX_{m\times n} \to dX^{D}_{i,m\times(1:n)}. \tag{4.1}$$

where

$$dX^{D}_{i,m\times(1:n)} = \begin{pmatrix} dx_{i1} & dx_{i2} & \cdots & dx_{in} \\ dx_{i2} & dx_{i3} & \cdots & dx_{i(n+1)} \\ \vdots & \vdots & \vdots & \vdots \\ dx_{im} & dx_{i(m+1)} & \cdots & dx_{i(m+n-1)} \end{pmatrix}, \tag{4.2}$$

where

$$dx_{ij} = x_{i(j+1)} - x_{ij}. \tag{4.3}$$

As above, let f_i still be elementary functions, so that a prediction model can be established for each row. For $j = 1$, we can establish a differential prediction model (referred to as the DD_1 model) that predicts one step ahead of schedule as follows:

$$\begin{cases} hA_{i,1(1:M)}X_{M\times M} = dX_{i(1:M)}, \\ \widehat{X}_{i(M+2)} = X_{i(M+1)} + hA_{i,1(1:M)}X_{i(M+1)}, \\ \widehat{X}_{i(M+k+1)} = \widehat{X}_{i(M+k)} + hA_{i,1(1:M)}\widehat{X}_{i(M+k)}, \ (k = 2, 3, \ldots). \end{cases} \tag{4.4}$$

where $i = 1, 2, ..., M$. We can also write Eq. (4.4) in a universal matrix form:

$$\begin{cases} hA_{(1:M),1(1:M)}X_{M\times M} = dX_{(1:M)\times(1:M)}, \\ \widehat{X}_{(1:M)(M+2)} = X_{(1:M)(M+1)} + hA_{(1:M),1(1:M)}X_{(1:M)(M+1)}, \\ \widehat{X}_{(1:M)(M+k+1)} = \widehat{X}_{(1:M)(M+k)} + hA_{(1:M),1(1:M)}\widehat{X}_{(1:M)(M+k)}, \\ (k = 2, 3, \ldots). \end{cases} \tag{4.5}$$

where

$$A_{(1:M)\times 2(1:M)} = \begin{pmatrix} a_{1,11} & a_{1,12} & \cdots & a_{1,1M} \\ a_{2,11} & a_{2,12} & \cdots & a_{2,1M} \\ \vdots & \vdots & \cdots & \vdots \\ a_{M,11} & a_{M,12} & \cdots & a_{M,1M} \end{pmatrix},$$

$$X_{M\times M} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1M} \\ x_{21} & x_{22} & \cdots & x_{2M} \\ \vdots & \vdots & \cdots & \vdots \\ x_{M1} & x_{M2} & \cdots & x_{MM} \end{pmatrix},$$

$$dX_{(1:M)\times(1:M)} = \begin{pmatrix} dx_{11} & dx_{12} & \cdots & dx_{1M} \\ dx_{22} & dx_{22} & \cdots & dx_{2M} \\ \vdots & \vdots & \cdots & \vdots \\ dx_{M2} & dx_{M3} & \cdots & dx_{MM} \end{pmatrix}.$$

For the general j, we can establish the following differential delay prediction model (referred to as the DD_j model) that predicts the j steps in advance:

$$\begin{cases} hA_{i,j(1:M)}X_{M\times M} = dX_{i(j:M+j-1)}, \\ \widehat{X}_{i(M+j+1)} = X_{i(M+j)} + hA_{i,j(1:M)}X_{i(M+1)}, \\ \widehat{X}_{i(M+j+k-1)} = \widehat{X}_{i(M+j+k-2)} + hA_{i,j(1:M)}\widehat{X}_{i(M+k)}, \ (k = 2, 3, \ldots). \end{cases} \tag{4.6}$$

where $i = 1, 2, ..., M$. We can write Eq. (4.6) in the following universal form:

$$\begin{cases} hA_{(1:M),j(1:M)}X_{M\times M} = dX_{(1:M)\times(j:M+j-1)}, \\ \widehat{X}_{(1:M)(M+j+1)} = X_{(1:M)(M+j)} + hA_{(1:M),j(1:M)}X_{(1:M)(M+1)}, \\ \widehat{X}_{(1:M)(M+j+k-1)} = \widehat{X}_{(1:M)(M+j+k-2)} + hA_{(1:M),j(1:M)}\widehat{X}_{(1:M)(M+k)}, \\ (k = 2, 3, \ldots). \end{cases} \tag{4.7}$$

where

$$A_{(1:M)\times 2(1:M)} = \begin{pmatrix} a_{1,j1} & a_{1,j2} & \cdots & a_{1,jM} \\ a_{2,j1} & a_{2,j2} & \cdots & a_{2,jM} \\ \vdots & \vdots & \cdots & \vdots \\ a_{M,j1} & a_{M,j2} & \cdots & a_{M,jM} \end{pmatrix},$$

$$dX_{(1:M)\times(j:M+J-1)} = \begin{pmatrix} dx_{1j} & dx_{1(J+1)} & \cdots & dx_{1(M+j-1)} \\ dx_{2j} & dx_{2(J+1)} & \cdots & dx_{2(M+j-1)} \\ \vdots & \vdots & \vdots & \vdots \\ dx_{Mj} & dx_{M(j+1)} & \cdots & dx_{M(M+j+1)} \end{pmatrix}.$$

To improve accuracy, for the DD_1 model (i.e. Eq. (4.4)), the following predictor-corrector differential delay prediction model (referred to as the $PCDD_1$ model) can be established to make predictions one step in advance:

$$\begin{cases} hA_{i,1(1:M)}X_{M\times M} = X_{i(1:M)}, \\ \overline{X}_{i(M+2)} = X_{i(M+1)} + hA_{i,1(1:M)}X_{i(M+1)}, \\ \widehat{X}_{i(M+2)} = X_{i(M+1)} + \frac{1}{2}A_{i,1(1:M)}(X_{i(M+1)} + \overline{X}_{i(M+2)}), \\ \overline{X}_{i(M+k+1)} = \widehat{X}_{i(M+k)} + hA_{i,1(1:M)}\widehat{X}_{i(M+k)}, \\ \widehat{X}_{i(M+k+1)} = \widehat{X}_{i(M+k)} + \frac{h}{2}A_{i,1(1:M)}(\widehat{X}_{i(M+k)} + \overline{X}_{i(M+k+1)}), \\ (k = 2, 3, \ldots). \end{cases} \tag{4.8}$$

where $i = 1, 2, ..., M$. If Eq. (4.8) is written in a more universal form, it would be as follows:

$$\begin{cases} hA_{(1:M),1(1:M)}X_{M\times M} = X_{(1:M)\times(1:M)}, \\ \overline{X}_{(1:M)(M+2)} = X_{(1:M)(M+1)} + hA_{(1:M),1(1:M)}X_{(1:M)(M+1)}, \\ \widehat{X}_{(1:M)(M+2)} = X_{(1:M)(M+1)} + \frac{h}{2}A_{(1:M),1(1:M)}(X_{(1:M)(M+1)} \\ \quad + \overline{X}_{(1:M)(M+2)}), \\ \overline{X}_{(1:M)(M+k+1)} = \widehat{X}_{(1:M)(M+k)} + hA_{(1:M),1(1:M)}\widehat{X}_{(1:M)(M+k)}, \\ \widehat{X}_{(1:M)(M+k+1)} = \widehat{X}_{(1:M)(M+k)} + \frac{h}{2}A_{(1:M),1(1:M)}(\widehat{X}_{(1:M)(M+k)} \\ \quad + \overline{X}_{(1:M)(M+k+1)}), \\ (k = 2, 3, \ldots). \end{cases} \tag{4.9}$$

Following the above, a predictor-corrector model $PCDD_j$ can be established to make predictions the j steps in advance.

► *Time-lag differential attractor mapping models*

Let $J \geq 0$, we construct the following time-lag delay mapping of differential attractor:

$$\Phi_N: hX_{([-J:1]m)\times n} \rightarrow dX^{D}_{i,m\times(1:(J+2)n)}. \tag{4.10}$$

As above, if f_i are still elementary functions, then

$$hA_{i,M\times([-J:1]M)}X_{([-J:1]M)\times(1:(J+2)M)} = dX^{D}_{i,M\times(1:(J+2)M)}. \tag{4.11}$$

where

$$A_{i,M\times([-J:1]M)} = (A_{i,M\times(1:M)}, A_{i,M\times(0:M-1)}, \dots, A_{i,M\times(-J:M-J-1)}), \tag{4.12}$$

$$\begin{aligned} X_{([-J:1]M)\times(1:(J+2)M)} &= (X_{M\times(1:(J+2)M)}, X_{M\times(0:(J+2)M-1)}, \\ &\quad \dots, X_{M\times(-J:(J+2)M-J-1)})^{\mathrm{T}} \\ &= \begin{pmatrix} X_{M\times(1:(J+2)M)} \\ X_{M\times(0:(J+2)M-1)} \\ \vdots \\ X_{M\times(-J:(J+2)M-J-1)} \end{pmatrix}, \end{aligned} \tag{4.13}$$

$$\begin{aligned} dX^{\mathrm{D}}_{i,M\times(1:(J+2)M)} &= (dX_{i(1:(J+2)M)}, dX_{i(2:(J+2)M+1)}, \\ &\quad \dots, dX_{i(M:(J+2)M+M-1)})^{\mathrm{T}} \\ &= \begin{pmatrix} dX_{i(1:(J+2)M)} \\ dX_{i(2:(J+2)M+1)} \\ \\ dX_{i(M:(J+2)M+M-1)} \end{pmatrix}. \end{aligned} \tag{4.14}$$

Each row can establish a predictive model. For $j = 1$, we establish the following time-lag differential delay prediction model (referred to as the TLDD_1 model) that utilizes the information from the previous $2 + J$ steps to make predictions 1 step in advance:

$$\begin{cases} hA_{i,1([-J:1]M)} X_{([-J:1]M)\times(1:(J+2)M)} = dX_{i(1:(J+2)M)}, \\ \widehat{X}_{i((J+2)M+2)} = X_{i((J+2)M+1)} + hA_{i,1([-J:1]M)} X_{([-J:1]M)((J+2)M)}, \\ \widehat{X}_{i((J+2)M+k+1)} = \widehat{X}_{i((J+2)M+k)} \\ \quad + hA_{i,1([-J:1]M)} \widehat{X}_{([-J:1]M)((J+2)M+k-1)}, \ (k = 2, 3, \dots). \end{cases} \tag{4.15}$$

where $i = 1, 2, \dots, M$. Eq. (4.15) can also be written in the following universal form:

$$\begin{cases} hA_{(1:M),1([-J:1]M)} X_{([-J:1]M)\times(1:(J+2)M)} = dX_{(1:M)\times(1:(J+2)M)}, \\ \widehat{X}_{(1:M)((J+2)M+2)} = X_{(1:M)((J+2)M+1)} + hA_{(1:M),1([-J:1]M)} X_{([-J:1]M)((J+2)M)}, \\ \widehat{X}_{(1:M)((J+2)M+k+1)} = \widehat{X}_{(1:M)((J+2)M+k)} \\ \quad + hA_{(1:M),1([-J:1]M)} \widehat{X}_{([-J:1]M)((J+2)M+k-1)}, \ (k = 2, 3, \dots). \end{cases} \tag{4.16}$$

For the general j, a time-lag differential delay prediction model (referred to as the TLDD_j model), which uses the information from the previous $2 + J$ steps to make predictions the j steps in advance, can be established as follows

$$\begin{cases} hA_{i,j([-J:1]M)} X_{([-J:1]M)\times(1:(J+2)M)} = X_{i(j:(J+2)M+j-1)}, \\ \widehat{X}_{i((J+2)M+j+1)} = X_{i((J+2)M+j)} + hA_{i,j([-J:1]M)} X_{([-J:1]M)((J+2)M)}, \\ \widehat{X}_{i((J+2)M+j+k)} = \widehat{X}_{i(M+j+k-1)} \\ \quad + hA_{i,1([-J:1]M)} \widehat{X}_{([-J:1]M)((J+2)M+k-1)}, \ (k = 2, 3, \dots). \end{cases} \tag{4.17}$$

where $i = 1, 2, \dots, M$. Eq. (4.17) can also be written in the following universal form:

$$\begin{cases} hA_{(1:M),j([-J:1]M)} X_{([-J:1]M)\times(1:(J+2)M)} = X_{(1:M)\times(j:(J+2)M+j-1)}, \\ \widehat{X}_{(1:M)((J+2)M+j+1)} = X_{(1:M)((J+2)M+j)} + hA_{(1:M),j([-J:1]M)} X_{([-J:1]M)((J+2)M)}, \\ \widehat{X}_{(1:M)((J+2)M+k+j)} = \widehat{X}_{(1:M)(M+j+k-1)} \\ \quad + hA_{(1:M),1([-J:1]M)} \widehat{X}_{([-J:1]M)((J+2)M+k-1)}, \ (k = 2, 3, \dots). \end{cases} \tag{4.18}$$

In order to improve accuracy, for the TLDD_1 model (i.e. Eq. (4.12)), a predictor-corrector time-lag differential delay model (referred to as the PCTLDD_1 model), which uses the information from the previous $2 + J$ steps to predict one step in advance, can be established:

$$\begin{cases} hA_{i,1([-J:1]M)} X_{([-J:1]M)\times(1:(J+2)M)} = dX_{i(1:(J+2)M)}, \\ \widehat{X}_{i((J+2)M+2)} = X_{i((J+2)M+1)} + hA_{i,1([-J:1]M)} X_{([-J:1]M)((J+2)M)}, \\ \widehat{X}_{i((J+2)M+2)} = X_{i((J+2)M+1)} + \dfrac{h}{2} A_{i,1([-J:1]M)} (X_{([-J:1]M)((J+2)M)} \\ \quad + \overline{X}_{i((J+2)M+2)}), \\ \overline{X}_{i((J+2)M+k+1)} = \widehat{X}_{i((J+2)M+k)} + hA_{i,1([-J:1]M)} X_{([-J:1]M)((J+2)M+k-1)}, \\ \widehat{X}_{i((J+2)M+k+1)} = \widehat{X}_{i((J+2)M+k)} \\ \quad + \dfrac{h}{2} A_{i,1([-J:1]M)} (\widehat{X}_{([-J:1]M)((J+2)M+k)} + \overline{X}_{i((J+2)M+k+1)}), \\ \quad (k = 2, 3, \dots). \end{cases} \tag{4.19}$$

where $i = 1, 2, \dots, M$. In addition, Eq. (4.19) can be written in the following universal form:

$$\begin{cases} hA_{(1:M),1([-J:1]M)} X_{([-J:1]M)\times(1:(J+2)M)} = dX_{(1:M)\times(1:(J+2)M)}, \\ \overline{X}_{(1:M)((J+2)M+2)} = X_{(1:M)((J+2)M+1)} \\ \quad + hA_{(1:M),1([-J:1]M)} X_{([-J:1]M)((J+2)M)}, \\ \widehat{X}_{(1:M)((J+2)M+2)} = X_{(1:M)((J+2)M+1)} + \\ \dfrac{h}{2} A_{(1:M),1([-J:1]M)} (X_{([-J:1]M)((J+2)M)} + \overline{X}_{(1:M)((J+2)M+2)}), \\ \overline{X}_{(1:M)((J+2)M+k+1)} = \widehat{X}_{(1:M)((J+2)M+k)} \\ \quad + hA_{(1:M),1([-J:1]M)} X_{([-J:1]M)((J+2)M+k-1)}, \\ \widehat{X}_{(1:M)((J+2)M+k+1)} = \widehat{X}_{(1:M)((J+2)M+k)} \\ + \dfrac{h}{2} A_{(1:M),1([-J:1]M)} (\widehat{X}_{([-J:1]M)((J+2)M+k)} + \overline{X}_{(1:M)((J+2)M+k+1)}), \\ \quad (k = 2, 3, \dots). \end{cases} \tag{4.20}$$

Similarly, following the previous approach, we can establish a predictor-corrector time-lag differential delay model PCTLDD_j that utilizes the information from the previous $2 + J$ steps to make predictions j steps in advance.

Time Reversal Models

When we discuss whether we can infer the past changes of a dynamical system based on its existing evolutionary data, we need to establish a backward mapping of the dynamical system over time. This type of model is termed as the time reversal model. This type of research is very interesting because it has the potential to provide a new method for reconstructing past changes of system, which is very useful for reconstructing paleoclimate data, etc. When establishing a time reversal model, we still need to establish a symbol system.

► *Basic notation*

Let i, j, k, m and n be integers, $j \le m$, $j \le n$, and let the backward raw vector be:

$$X_{i(m:j)} = (x_{im}, x_{i(m-1)}, \dots, x_{ij}), \tag{5.1}$$

$$A_{i,k(m:j)} = (a_{i,km}, a_{i,k(m-1)}, \dots, a_{i,kj}), \tag{5.2}$$

The backward column vector is:

$$X_{(m:j)i} = (x_{mi}, x_{(m-1)i}, \dots, x_{ji})^{\mathrm{T}}, \tag{5.3}$$

$$A_{i,(m:j)k} = (a_{i,mk}, a_{i,(m-1)k}, \dots, a_{i,jk})^{\mathrm{T}}, \tag{5.4}$$

The backward evolution matrix is denoted as:

$$\begin{aligned} X_{m\times(n:j)} = X_{(1:m)\times(n:j)} &= \begin{pmatrix} x_{1n} & x_{1(n+1)} & \cdots & x_{1j} \\ x_{2n} & x_{2(n+1)} & \cdots & x_{2j} \\ \vdots & \vdots & \vdots & \vdots \\ x_{mn} & x_{m(n+1)} & \cdots & x_{mj} \end{pmatrix} \\ &= (X_{1(n:j)}, X_{2(n:j)}, \dots, X_{m(n:j)})^{\mathrm{T}} \\ &= (X_{(1:m)n}, X_{(1:m)(n-1)}, \dots, X_{(1:m)j}). \end{aligned} \tag{5.5}$$

When $j = 1$, the backward evolution matrix is denoted as:

$$\begin{aligned} X_{m\times(n\,:\,j)} = X_{(1\,:\,m)\times(n\,:\,1)} &= \begin{pmatrix} x_{1n} & x_{1(n-1)} & \cdots & x_{11} \\ x_{2n} & x_{2(n-1)} & \cdots & x_{21} \\ \vdots & \vdots & \vdots & \vdots \\ x_{mn} & x_{m(n-1)} & \cdots & x_{m1} \end{pmatrix} \\ &= (X_{1(n\,:\,1)}, X_{2(n\,:\,1)}, \cdots, X_{m(n\,:\,1)})^{\mathrm{T}} \\ &= (X_{(1\,:\,m)n}, X_{(1\,:\,m)(n-1)}, \cdots, X_{(1\,:\,m)j}) \end{aligned} \tag{5.6}$$

The univariate backward extension matrix is denoted as:

$$\begin{aligned} X^{-\mathrm{D}}_{i,m\times(n-1:j)} = X^{-\mathrm{D}}_{i,(1:m)\times(n-1:j)} &= \begin{pmatrix} x_{i(n-1)} & x_{i(n-2)} & \cdots & x_{ij} \\ x_{i(n-2)} & x_{i(n-3)} & \cdots & x_{i(j-1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{i(n-m)} & x_{i(n-m-1)} & \cdots & x_{i(j-m+1)} \end{pmatrix} \\ &= (X_{i(n-1:j)}, X_{i(n-1:j-1)}, \dots, X_{i(n-m:j-m+1)})^{\mathrm{T}} \\ &= (X^{\mathrm{T}}_{i(n-1:n-m)}, X^{\mathrm{T}}_{i(n-2:n-m-1)}, \dots, X^{\mathrm{T}}_{i(j:j-m+1)}). \end{aligned} \tag{5.7}$$

When $j = 0$, the univariate backward extension matrix is denoted as:

$$\begin{aligned} X^{-\mathrm{D}}_{i,m\times(n-1:0)} = X^{-\mathrm{D}}_{i,(1:m)\times(n-1:0)} &= \begin{pmatrix} x_{i(n-1)} & x_{i(n-2)} & \cdots & x_{i0} \\ x_{i(n-2)} & x_{i(n-3)} & \cdots & x_{i(-1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{i(n-m)} & x_{i(n-m-1)} & \cdots & x_{i(-m+1)} \end{pmatrix} \\ &= (X_{i(n-1:0)}, X_{i(n-2:-1)}, \dots, X_{i(n-m:-m+1)})^{\mathrm{T}} \\ &= (X^{\mathrm{T}}_{i(n-1:n-m)}, X^{\mathrm{T}}_{i(n-2:n-m+1)}, \dots, X^{\mathrm{T}}_{i(0:-m+1)}). \end{aligned} \tag{5.8}$$

The backward coefficient matrix is denoted as:

$$\begin{aligned} A_{i,m\times(m:1)} = A_{i,(1:m)\times(m:1)} &= \begin{pmatrix} a_{i,1m} & a_{i,1(m-1)} & \cdots & a_{i,11} \\ a_{i,2m} & a_{i,2(m-1)} & \cdots & a_{i,21} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i,mm} & x_{i,m(m-1)} & \cdots & a_{i,m1} \end{pmatrix} \\ &= (A_{i,1\times(m:1)}, A_{i,2\times(m:1)}, \dots, A_{i,m\times(m:1)})^{\mathrm{T}} \\ &= (A_{i,1\times(m:1)m}, A_{i,(1:m)(m-1)}, \dots, A_{i,(1:m)1}). \end{aligned} \tag{5.9}$$

The general backward coefficient matrix is:

$$\begin{aligned} A_{i,m\times(n:j)} = A_{i,(1:m)\times(n:j)} &= \begin{pmatrix} a_{i,1n} & a_{i,1(n-1)} & \cdots & a_{i,1j} \\ a_{i,2n} & a_{i,2(mn-1)} & \cdots & a_{i,2j} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i,mn} & x_{i,m(n-1)} & \cdots & a_{i,mj} \end{pmatrix} \\ &= (A_{i,1\times(n:j)}, A_{i,2\times(n:j)}, \dots, A_{i,m\times(n:j)})^{\mathrm{T}} \\ &= (A_{i,(1:m)n}, A_{i,(1:m)(n-1)}, \dots, A_{i,(1:m)j}). \end{aligned} \tag{5.10}$$

► *Conventional time reversal models*

Construct the following time reversal mapping:

$$\Phi^*_N : X_{m\times(N:1)} \longrightarrow X^{-\mathrm{D}}_{i,M\times(N-1:0)}, \tag{5.11}$$

where Φ^*_N is a nonlinear mapping, that is

$$\begin{pmatrix} x_{1N} & x_{1(N-1)} & \cdots & x_{11} \\ x_{2N} & x_{2(N-1)} & \cdots & x_{21} \\ \vdots & \vdots & \vdots & \vdots \\ x_{MN} & x_{M(N-1)} & \cdots & x_{M1} \end{pmatrix} \xrightarrow{\Phi^*_N} \begin{pmatrix} x_{i(N-1)} & x_{i(N-2)} & \cdots & x_{i0} \\ x_{i(N-2)} & x_{i(N-3)} & \cdots & x_{i(-1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{i(N-M)} & x_{i(N-M-1)} & \cdots & x_{i(-M+1)} \end{pmatrix}. \tag{5.12}$$

In this way, by constructing Φ^*_N, it is possible to recover the past of the variable $x_i (i = 1, 2, \dots, m)$ using existing data. Similar to the previous discussion, we can establish the following backtracking reconstruction model (referred to as the D_{-1} model, i.e., backtracking by one step):

$$\begin{cases} A_{i,1(M:1)} X_{i\times(M:1)} = X_{i(M-1:0)}, \\ \widehat{X}_{i(-1)} = A_{i,1(M:1)} X_{i0}, \\ \widehat{X}_{i(-k)} = A_{i,1(M:1)} \widehat{X}_{i(-k+1)}, \ (k = 2, 3, \dots). \end{cases} \tag{5.13}$$

where $i = 1, 2, \dots, M$. In addition, Eq. (5.13) can be written in the following universal form:

$$\begin{cases} A_{(1:M),1(M:1)} X_{(1:M)\times(M:1)} = X_{(1:M)\times(M-1:0)}, \\ \widehat{X}_{(1:M)(-1)} = A_{(1:M),1(M:1)} X_{(1:M)0}, \\ \widehat{X}_{(1:M)(-k)} = A_{(1:M),1(M:1)} \widehat{X}_{(1:M)(-k+1)}, \ (k = 2, 3, \dots). \end{cases} \tag{5.14}$$

where

$$A_{(1:M),1(M:1)}=\begin{pmatrix} a_{1,1M} & a_{1,1(M-1)} & \cdots & a_{1,11} \\ a_{2,1M} & a_{2,1(M-1)} & \cdots & a_{2,11} \\ \vdots & \vdots & \vdots & \vdots \\ a_{M,1M} & x_{M,1(M-1)} & \cdots & a_{M,11} \end{pmatrix},$$

$$X_{(1:M)\times(M:1)}=\begin{pmatrix} x_{1M} & x_{1(M-1)} & \cdots & x_{11} \\ x_{2M} & x_{2(M-1)} & \cdots & x_{21} \\ \vdots & \vdots & \vdots & \vdots \\ x_{MM} & x_{M(M-1)} & \cdots & x_{M1} \end{pmatrix},$$

$$X_{(1:M)\times(M-1:0)}=\begin{pmatrix} x_{1(M-1)} & x_{1(M-2)} & \cdots & x_{10} \\ x_{2(M-1)} & x_{2(M-2)} & \cdots & x_{20} \\ \vdots & \vdots & \vdots & \vdots \\ x_{M(M-1)} & x_{M(M-2)} & \cdots & x_{M0} \end{pmatrix}.$$

In addition, the following backtracking reconstruction model (referred to as the D_{-j} model), which backtracks $j(j>1)$ steps, can be established as follows

$$\begin{cases} A_{i,j(M:1)}X_{M\times(M:1)}=X_{i(M-j:1-j)}, \\ \widehat{X}_{i(-j)}=A_{i,j(M:1)}X_{i0}, \\ \widehat{X}_{i(-j-k+2)}=A_{i,j(M:1)}\widehat{X}_{i(-k+2)},\ (k=2,3,\ldots). \end{cases} \quad (5.15)$$

where $i=1, 2, ..., M$. Eq. (5.15) can be written in the following universal form:

$$\begin{cases} A_{(1:M),j(M:1)}X_{M\times(M:1)}=X_{(1:M)\times(M-j:1-j)}, \\ \widehat{X}_{(1:M)(-j)}=A_{(1:M),j(M:1)}X_{(1:M)0}, \\ \widehat{X}_{(1:M)(-j-k+2)}=A_{(1:M),j(M:1)}\widehat{X}_{(1:M)(-k+2)},\ (k=2,3,\ldots). \end{cases} \quad (5.16)$$

where

$$A_{(1:M),j(M:1)}=\begin{pmatrix} a_{1,jM} & a_{1,j(M-1)} & \cdots & a_{1,j1} \\ a_{2,jM} & a_{2,j(M-1)} & \cdots & a_{2,j1} \\ \vdots & \vdots & \vdots & \vdots \\ a_{M,jM} & x_{M,j(M-1)} & \cdots & a_{M,j1} \end{pmatrix}$$

$$X_{(1:M)\times(M-j:1-j)}=\begin{pmatrix} x_{1(M-j)} & x_{1(M-j-1)} & \cdots & x_{1(1-j)} \\ x_{2(M-j)} & x_{2(M-j-1)} & \cdots & x_{2(1-j)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{M(M-j)} & x_{M(M-j-1)} & \cdots & x_{M(1-j)} \end{pmatrix}$$

▶ *Time-lag time reversal models*

Let $J\geq 1$, we construct the following time-lag time reversal mapping:

$$\Phi_N^{*}:X_{([1:1+J]M)\times(N:1)}\rightarrow X^{-D}_{i,M\times((J+1)(N-1):0)}. \quad (5.17)$$

This indicates that the information from the following $1+J$ steps also needs to be considered. Similar to the discussion above, one can have the following equation

$$A_{i,M\times([1:1+J]M)}X_{([1:1+J]M)\times((J+1)M:1)}=X_{i,M\times((J+1)M-1:0)}. \quad (5.18)$$

where

$$A_{i,M\times([1:1+J]M)}=(A_{i,M\times([M:1)},A_{i,M\times([M+1:2)},\ldots,A_{i,M\times([M+J-1:J)}), \quad (5.19)$$

$$X_{([1:1+J]M)\times((J+1)M:1)}=(X_{M((J+1)M:1)},X_{M\times((J+1)M+1:2)},\ldots,X_{M\times((J+1)M+J-1:J)})^{\mathrm{T}}$$
$$=\begin{pmatrix} X_{M\times((J+1)M:1)} \\ X_{M\times((J+1)M+1:2)} \\ \vdots \\ X_{M\times((J+1)M+J-1:J)} \end{pmatrix} \quad (5.20)$$

$$X_{i,M\times([J+1]M-1:0)}=(X_{i(0:(J+1)M-1)},X_{i(1:(J+1)M)},\ldots,X_{i(M-1:(J+1)M+M-2)}). \quad (5.21)$$

Firstly, a time-lag backtracking reconstruction model (referred to as the $TLD_{-1}(1+J)$ model), which take one step backtracking from the information of the following $1+J$ steps, can be established as follows

$$\begin{cases} A_{i,1([1:1+J]M)}X_{([1:1+J]M)\times([J+1]M:1)}=X_{i((J+1)M-1:0)}, \\ \widehat{X}_{i(-1)}=A_{i,1([1:1+J]M)}X_{([1:1+J]M)0}, \\ \widehat{X}_{ik}=A_{i,1([1:1+J]M)}\widehat{X}_{([1:1+J]M)(k+1)},\ (k=-2,-3,\ldots). \end{cases} \quad (5.22)$$

where $i=1, 2, ..., M$,

$$A_{i,1([1:1+J]M)}=(A_{i,1(M:1)},A_{i,1(M+1:2)},\cdots,A_{i,1(M+J-1:J)}),$$

$$X_{i((J+1)M-1:0)}=(x_{i,((J+1)M-1)},x_{i((J+1),M-2)},\cdots,x_{i0}).$$

If it is written in general matrix form, Eq. (5.22) becomes:

$$\begin{cases} A_{(1:M),1([-J:1]M)}X_{([-J:1]M)\times([J+1]M:1)}=X_{(1:M)\times((J+1)M-1:0)}, \\ \widehat{X}_{(1:M)(-1)}=A_{(1:M),1([-J:1]M)}X_{([-J:1]M)0}, \\ \widehat{X}_{(1:M)k}=A_{(1:M),2([-J:1]M)}\widehat{X}_{([-J:1]M)(k+1)},\ (k=-2,-3,\ldots). \end{cases} \quad (5.23)$$

where

$$X_{(1:M),1([1:1+J]M)}=\begin{pmatrix} A_{1,1(M:1)} & A_{1,1(M+1:2)} & \cdots & A_{1,1(M+J-1:J)} \\ A_{2,1(M:1)} & A_{2,1(M+1:2)} & \cdots & A_{2,1(M+J-1:J)} \\ \vdots & \vdots & \vdots & \vdots \\ A_{M,1(M:1)} & A_{M,1(M+1:2)} & \cdots & A_{M,1(M+J-1:J)} \end{pmatrix},$$

$$X_{(1:M)\times((J+1)M-1:0)}=\begin{pmatrix} x_{1((J+1)M-1)} & x_{1((J+1)M-2)} & \cdots & x_{10} \\ x_{2((J+1)M-1)} & x_{2((J+1)M-2)} & \cdots & x_{20} \\ \vdots & \vdots & \vdots & \vdots \\ x_{M((J+1)M-1)} & x_{M((J+1)M-2)} & \cdots & x_{M0} \end{pmatrix}.$$

When $j>1$, we establish a time-lag backtracking reconstruction model (referred to as the $TLD_{-j}(1+J)$ model) that uses the information of the following $(1+J)$ steps to make backtracking steps in advance

$$\begin{cases} A_{i,j([1:1+J]M)}X_{([1:1+J]M)\times([J+1]M:1)}=X_{i((J+1)M-1-j:1-j)}, \\ \widehat{X}_{i(-j)}=A_{i,j([1:1+J]M)}X_{([1:1+J]M)0}, \\ \widehat{X}_{i(k-j)}=A_{i,j([1:1+J]M)}\widehat{X}_{([1:1+J]M)(k+1-j)},\ (k=-2,-3,\ldots). \end{cases} \quad (5.24)$$

where $i=1, 2, ..., M$,

$$A_{i,j([1:1+j]M)} = (A_{i,j(M:1)}, A_{i,j(M+1:2)}, \dots, A_{i,j(M+j-1:j)}),$$

$$X_{i(J+1)M-1-j:1-j)} = (x_{i(j+1)M-1-j}, x_{i(J+1)M-2-j)}, \dots, x_{i(1-j)}).$$

If it is written in general matrix form, Eq. (5.24) becomes:

$$\begin{cases} A_{(1:M),j([1:1+J]M)} X_{([1:1+J]M)\times((J+1)M:1)} = X_{(1:M)\times((J+1)M-1-j:1-j)}, \\ \widehat{X}_{i(-j)} = A_{(1:M),j([1:1+J]M)} X_{([1:1+J]M)0}, \\ \widehat{X}_{(1:M)(k-j)} = A_{(1:M),j([1:1+J]M)} \widehat{X}_{([1:1+J]M)(k+1-j)}, \\ (k = -2, -3, \dots). \end{cases} \tag{5.25}$$

$$A_{(1:M),j([1:1+J]M)} = \begin{pmatrix} A_{1,j(M:1)} & A_{1,j(M+1:1)} & \cdots & A_{1,j(M+J-1:J)} \\ A_{2,j(M:1)} & A_{2,j(M+1:1)} & \cdots & A_{2,j(M+J-1:J)} \\ \vdots & \vdots & \vdots & \vdots \\ A_{M,j(M:1)} & A_{M,j(M+1:1)} & \cdots & A_{M,j(M+J-1:J)} \end{pmatrix}, \tag{5.26}$$

$$X_{(1:M)((J+1)M-1-j:1-j)} = \begin{pmatrix} x_{1((J+1)M-1-j)} & x_{1((J+1)M-2-j)} & \dots & x_{1(1-j)} \\ x_{2((J+1)M-1-j)} & x_{2((J+1)M-2-j)} & \cdots & x_{2(1-j)} \\ \vdots & \vdots & \vdots & \vdots \\ x_{M((J+1)M-1-j)} & x_{M((J+1)M-2-j)} & \cdots & x_{M(1-j)} \end{pmatrix}. \tag{5.27}$$

For the time reversal model, it is also possible to consider introducing the time-lag information to modelling.

Conclusion and Discussion

In this paper we introduce the framework of the modelling theory of the DSDL. Firstly, on the basis of attractor theory, we construct the linear and conventional nonlinear prediction models of the DSDL by introducing the dynamical evolution mapping relationship between non-delay and delay attractors of dynamical systems. Meanwhile, the corresponding predictor-corrector models are presented. We introduce the concept of key variables for interpretability of the model, and propose two ways for how to implement the set of key variables. Secondly, to take time-lag information into account, we construct the time-lag mapping between previous information evolution matrix and univariate delay matrix, and thus establish the time-lag models and corresponding predictor-corrector models as well. Furthermore, we establish the nonlinear extension relationships between non-delay attractor and differential attractor. As a result, the differential attractor mapping models are established, which contain the conventional differential attractor mapping models, time-lag differential attractor mapping models and corresponding predictor-corrector models. Besides, it is also a very important issue to infer the past from the present of a dynamical system. For this, therefore, we construct the time reversal mapping between the backward evolution matrix and univariate backward extension matrix, and in turn, build the time reversal models, which contain the conventional time reversal models and time-lag time reversal models.

In fact, there are essential differences among dynamical, statistical, ML and DSDL models, as shown in Fig. 2. Both statistical model and ML model cannot guarantee the consistency between model attractor and the attractor of the original dynamical system. Practice (Wang and Li, 2024a, b; Li et al., 2025; Wu et al., 2025) has proven that the DSDL, which combines nonlinear dynamics theory with deep learning technology, effectively enhances the predictive capability of chaotic dynamical systems and achieves transparency in prediction models. Compared with state-of-art ML or deep learning methods, the DSDL demonstrates its significant superiority and excellent ability in purifying polluted information. In addition, the DSDL solves the “black box” problem and opens up a new path for constructing more reliable and transparent prediction models.

Acknowledgements

This work was jointly supported by National Natural Science Foundation of China (42288101), Laoshan Laboratory (No.LSKJ202202600), Shandong Natural Science Foundation Project (ZR2019ZD12), and Fundamental Research Funds for the Central Universities (202242001).

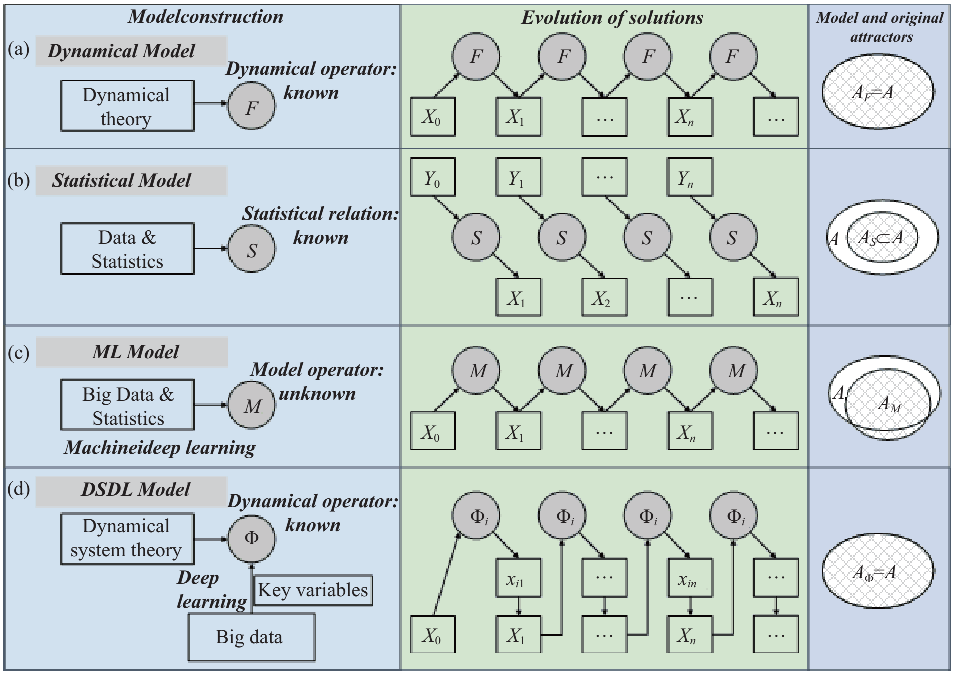

Fig. 2 Differences among dynamical, statistical, ML and DSDL models. A, A_F, A_S, A_M and A_Φ represent the attractor sets of original system, dynamical model, statistical model, AI model and DSDL model, respectively

References

1. Eftekhari, A., H. L. Yap, M. B. Wakin and C. J. Rozell., 2018: Stabilizing embedology: geometry-preserving delay-coordinate maps. P*hys. Rev. E*, **97**: 022222.
2. Komalapriya, C., M. Thiel, M. C. Romano, N. Marwan, U. Schwarz, and J. Kurths, 2008: Reconstruction of a system's dynamics from short trajectories. *Phys. Rev. E*, **78**: 066217.
3. Komalapriya, C., M. C. Romano, M. Thiel, N. Marwan, J. Kurths, I. Z. Kiss, and J. L. Hudson, 2010: An automated algorithm for the generation of dynamically reconstructed trajectories. *Chaos*, **20**: 013107.
4. Li, H., J. P. Li, Z. X. Wu, M. Y. Wang, G. C. Liu, R. P. Sun, R. Z. Li, N. Wang, H. B. Song and S. X. Zhen, 2025: Dynamics-based predictions of infinite-dimensional complex systems using Dynamical System Deep Learning method. *Mach. Learn.: Sci. Technol.*, **online**, https://doi.org/10.1088/2632-2153/adc53b.
5. Li, J. P., 1997: Qualitative theory of the dynamical equations of atmospheric and oceanic motion and its applications. PhD Thesis, Lanzhou University, Lanzhou.
6. Li, J. P. and J. F. Chou, 1997: Existence of the atmosphere attractor. *Sci. China Earth Sci*. **40**: 215−224.
7. Li, J. P. and J. F. Chou, 1998: The qualitative theory on the dynamical equations of atmospheric motion and its applications. *Chinese J. Atmos. Sci*. **22**: 348−360.
8. Li, J. P. and J. F. Chou, 2003: Global analysis theory of climate system and its applications. *Chin. Sci. Bull.* **48**: 1034−1039.
9. Lorenz, E. N., 1963: Deterministic nonperiodic flow. *J. Atmos. Sci*. **20**: 130−141.
10. Ma, H., T. Zhou, K. Aihara, and L. Chen, 2014: Predicting time series from short-term high-dimensional data. *Int. J. Bifurcat. Chaos*, **24**: 143003.
11. Ma, H., S. Leng, K. Aihara, W. Lin W and L. Chen, 2018: Randomly distributed embedding making short-term high-dimensional data predictable. *Proc. Natl. A Sci. U. S. A.*, **115**: 9994–10002.
12. Robinson, J., 2005: A topological delay embedding theorem for infinite-dimensional dynamical systems. *Nonlinearity*, **18**: 2135–2143.
13. Takens, F., 1981: Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence*, Warwick 1980, vol. 898 (eds. Rand, D. A. and Young, L.-S.), Springer, 366–381.

14. Wang, M. Y. and J. P. Li, 2024a: Interpretable predictions of chaotic dynamical systems using dynamical system deep learning. *Sci. Rep*. **14**: 3143.
15. Wang, M. Y. and J. P. Li, 2024b: Exploring the potential of contemporary deep learning methods in purifying polluted information. *Mach. Learn.: Sci. Technol*. **5**: 045026.
16. Wu, Z. X., J. P. Li, H. Li, M. Y. Wang, N. Wang, G. C. Liu, 2025: Robust prediction of chaotic systems with random errors using dynamical system deep learning. *Mach. Learn.: Sci. Technol*., **online**,
17. Yap, H. L., A. Eftekhari, M. B. Wakin and C. J. Rozell., 2014: A first analysis of the stability of Takens' embedding. *2014 IEEE Global Conference on Signal and Information Processing* (GlobalSIP), pp 404−408.

谢素娟
教授

谢素娟，博士，女，现为中国海洋大学管理学院会计学教授、博士生导师，入选山东省泰山学者青年专家、中国海洋大学青年英才工程第一层次，*Sustainability Accounting，Management and Policy Journal* 副主编、*Marine Development* 编委、中国技术经济协会海洋技术经济分会副秘书长、中国成本研究会理事、山东省管理学学会理事、山东省企业管理研究会理事、广州市社会组织专家库成员。主要研究领域为高管激励、国资国企改革、可持续发展会计。在领域内权威期刊发表研究成果 10 余篇，发表于 *Journal of Corporate Finance*；*Emerging Markets Review*；*Entrepreneurship，Theory and Practice*；*Accounting & Finance*，《管理科学学报》等刊物。

Dr. Xie Sujuan，is currently a Professor of Accounting and Doctoral Supervisor at the School of Management，Ocean University of China. She is a recipient of the Young Expert of Shandong Province' s Taishan Scholar Program and a top-tier talent in the university' s Youth Talent Engineering Initiative. Dr. Xie serves as an Associate Editor of the *Sustainability Accounting，Management and Policy Journal*，an Editorial Board Member of *Marine Development*，Deputy Secretary of the Marine Technology and Economics Division of the China Association of Technological Economics，a Council Member of the China Cost Research Society，a Council Member of the Shandong Management Science Association，a Council Member of the Shandong Enterprise Management Research Association，and an Expert in the Guangzhou Social Organization Expert Database.

Her primary research interests include executive incentives，SOE reform，and sustainable development accounting. Dr. Xie has published over ten research articles in leading journals such as *Journal of Corporate Finance*，*Emerging Markets Review*，*Entrepreneurship Theory and Practice*，*Accounting & Finance*，and *Journal of Management Science in China*.

孟昭苏
副教授

孟昭苏，女，1983 年 11 月生，澳大利亚新南威尔士大学海洋学博士。中国海洋大学经济学院副教授。具有经济学和管理学、海洋科学多学科交叉的学术背景，主要从事气候经济、海洋经济、金融市场等方向的研究。主持国家社科基金一般项目、山东省社科基金青年项目等；参与国家社科基金重大项目、重点项目、一般项目，参与国家自然科学基金项目、国家重点研发计划、教育部人文社科发展报告项目、国家公益项目子任务等 10 余项课题研究。参与编写著作 6 部。在 *Ocean & Coastal Management*，*Frontiers in Environmental Science*，*Frontiers in Environmental Science*，*Grey System: Theory and Application* 等期刊发表论文多篇。

Dr. Zhaosu Meng is an Associate Professor at the School of Economics, Ocean University of China. She received her Ph. D. in Marine Science from the University of New South Wales, Australia. With an interdisciplinary background spanning economics, management, and marine science, her research focuses on climate economics, marine economics, and financial markets.

Dr. Meng has led several major research projects, including those funded by the National Social Science Foundation of China and the Shandong Province Social Science Youth Fund. She has participated in over ten national-level research initiatives, including key projects from the National Natural Science Foundation and the National Key R&D Program. Her work has been published in leading journals such as Ocean & Coastal Management and Frontiers in Environmental Science, and she has contributed to six books in her field.