Dynamic Mode Decomposition with Memory

Ryoji Anzaki*, Kei Sano†, Takuro Tsutsui, Masato Kazui, Takahito Matsuzawa

AI Development Department, System Development Center, Tokyo Electron Ltd., Sapporo City 060-0003, Japan

October 19, 2022

Abstract

In many realms of science and engineering, the time-evolution of a system plays a key role in analyzing, controlling and predicting the behavior of the system. Ranging from the climate data collected by the observation satellites to the classical mechanical motion of manufacturing equipment, there are vast amount of accumulated time-series data in ready-to-use formats.

One established and still developing method for time-series data analysis is the dynamic mode decomposition (DMD), a linear operation-based, model-free method proposed in 2008 by Schmid *et al.* The DMD, especially in the context of the Koopman theory, assumes the time-evolution model in which the variables evolve in time according to a homogeneous, constant-coefficient first-order ordinary differential equation (ODE). Although the DMD proves itself as one of the successful methods for time-series data analysis in its relatively short history, the abovementioned limitations on the time-evolution model still restricts the DMD from application to wider range of phenomena. Among such phenomena, we pay attention to the memory effects in time-series data. A system is said to have memory if its time-evolution is determined by the past states of the system. The DMD is based on the ODEs and thus cannot incorporate the memory effects.

In this paper, we present a novel method, DMD with memory (DMDm), to overcome the *memoryless* restriction on the time-evolution model in the existing DMD methods. We introduce a class of initial value problems whose solutions correspond to the eigenmode of a linear operation. Using such a linear transformation in time domain, instead of the time-difference operator, we enlarge the DMD into a wider class of time-evolution models.

As a solid example of this idea, we utilize the Caputo fractional differential to extend the DMD so that one can analyze the time-series data with power-low memory effects, which is seen in various phenomena e.g., viscoelastic matter, fluid dynamics with surface effects, and the mechanical slider with grease. We thus developed *fractional* DMD, a DMD-based method with arbitrary (real-value) order differential operations. We want to emphasize the fact that you can actually *optimize* the order of the equation with a standard optimization algorithm, so that one get the order of the equation that best explains the nature of the time-series data. This is also true for any DMDm method with a parameterized time-evolution model. We demonstrated the results for the synthetic data, and successfully estimated the model parameters.

The proposed method is expected to be useful not only for the scientific use, but also for model estimation, control and failure detection of mechanical, thermal and fluid systems in factory machines, such as in modern semiconductor manufacturing equipment.

Keywords: dynamic mode decomposition, fractional-order derivative, time-series data

*Electronic Address: ryoji.anzaki@tel.com
†Electronic Address: kei.sano@tel.com

1 Introduction

Dynamic mode decomposition (DMD) [16, 23, 2, 17] is a model-free, linear algebra-based method for time-series data analysis. First developed by Schmid for analysis of experimental data in fluid dynamics [16], the DMD becomes popular among wide variety of scientific realms, ranging from climatology [10, 15], plasma physics [14] to dissipative quantum systems [12], as well as applications in fluid dynamics [7]. Besides its success in data-driven science, the mathematical structure of the DMD is also attracting wide attention, especially in terms of its connection with the Koopman theory [6, 11, 2].

1.1 Algorithm of DMD

We show the algorithm of DMD in its simplest form, the *exact* DMD [23, 17]. Suppose we have a sequence of time points $\mathcal{T} = \{t_0, t_1, ..., t_{m-1}\}$ with $t_j > t_i$ for $j > i$. Suppose we have a collection of data pairs, such as instantaneous observations of a system, $\{\boldsymbol{x}_0, ..., \boldsymbol{x}_{m-1}\}$. Each *observation* $\boldsymbol{x}_k \in \mathbf{R}^n (k = 0, ..., m-1)$ is a state (column) vector at time t_k. The exact DMD is formulated as the least-squares method for $\dot{\boldsymbol{x}}_k$ with the time-evolution model in continuous time,

$$\dot{\boldsymbol{x}}(t) = \mathcal{A}\boldsymbol{x}(t), \tag{1}$$

with $\mathcal{A} \in \mathbf{R}^{n\times n}$ being the parameter of continuous-time time-evolution model. For a uniformly discretized $\mathcal{T}$ with time interval $\Delta t > 0$, the optimization of $\mathcal{A}$ is performed by a matrix manipulation by letting $X = \left[\boldsymbol{x}_0, \boldsymbol{x}_1, ..., \boldsymbol{x}_{m-2}\right]$ and $X' = \left[\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_{m-1}\right]$, as

$$\mathcal{A} \simeq \frac{A - \hat{1}_n}{\Delta t}, \quad A = \arg\min_{A'} \|X' - A'X\|_F = X'X^+, \tag{2}$$

where $\hat{1}_n$ is the $n \times n$ unit matrix, and $\| \bullet \|_F$ is the matrix Frobenius norm. Note that the matrix A is the transition matrix: $A\boldsymbol{x}_k = \boldsymbol{x}_{k+1}$. The Moore-Penrose pseudo-inverse $\bullet^+$ can be calculated by the singluar-value decomposition (SVD) $X = U\Sigma V^*$. Here $U \in \mathbf{C}^{n\times n}$ and $V \in \mathbf{C}^{m\times m}$ are unitary, i.e., $U^*U = \hat{1}_n$ and $V^*V = \hat{1}_m$, where $\bullet^*$ denotes the adjoint matrix. $\Sigma \in \mathbf{R}^{n\times m}$ contains the singular values of X. The pseud-inverse X^+ is now calculated as $X^+ = V\Sigma^{-1}U^*$, so that the parameter of the discrete-time time evolution model is now given by,

$$A = X'V\Sigma^{-1}U^*. \tag{3}$$

The rank-r approximation of the SVD for a positive integer $r < n$ becomes $X \simeq U_r\Sigma_r V_r^*$ for $U_r \in \mathbf{C}^{n\times r}, \Sigma_r \in \mathbf{C}^{r\times r}$ and $V_r \in \mathbf{C}^{n\times r}$. Thus we can get the rank-r representation of the discrete-time time-evolution model as follows:

$$A_r = U_r^* A U_r = U_r^* X' V_r \Sigma_r^{-1}. \tag{4}$$

The low-rank representation is often used for data with large dimensions.

1.2 Time-evolution model

The time-evolution model of the existing DMD is the first-order ordinary differential equations (ODEs) 1, and the generic solution is expressed as the superposition of the DMD modes (i.e., eigenvectors of the coefficient matrix $\mathcal{A}$) with time-evolution function expressed in exponential of time:

$$\boldsymbol{x}(t) = \sum_{j=0}^{n-1} \vec{\phi}_j \exp(\omega_j t) b_j = \boldsymbol{\Phi} \exp(\boldsymbol{\Omega} t)\boldsymbol{b}, \tag{5}$$

with $\boldsymbol{\Phi} = \left[\vec{\phi}_0, \vec{\phi}_1, ..., \vec{\phi}_{n-1}\right]$ and $\boldsymbol{\Omega} = \mathrm{diag}(\omega_0, \omega_1, ..., \omega_{n-1})$ being the eigenvectors and eigenvalues of $\mathcal{A}$, respectively. The vector $\vec{\phi}_j$ is the j-th DMD mode, corresponding to the DMD eigenvalue ω_j, while

b_j is the loadings for each DMD mode. Note that the DMD modes and the exponential function share the same eigenvalues to satisfy Equation 1. A low-rank representation corresponds to the replacement $\sum_{j=0}^{n-1} \to \sum_{j=0}^{r-1}$ in Equation 5.

Using the generic solution above, the time-evolution model Equation 1 can be solved for a given initial value $\boldsymbol{x}_0$, via the replacement $\boldsymbol{b} = \boldsymbol{\Phi}^{+}\boldsymbol{x}_0$ in Equation 5.

1.3 Non-uniform time points

The sequence of time points $\mathcal{T}$ is uneven in many realistic situations. In such cases, one can employ the discrete representation of the time-derivative operator. Hereafter, let us assume that the data matrix X is defined by $X = [\boldsymbol{x}_0, \boldsymbol{x}_1, ..., \boldsymbol{x}_{m-1}]$. The time-evolution model now becomes of the form,

$$XD_m^{(1)} = AX_{1:|0}, \tag{6}$$

with

$$X_{k:|a} = [\overbrace{\boldsymbol{a}, ..., \boldsymbol{a}}^{k}, \boldsymbol{x}_k, \boldsymbol{x}_{k+1}, ..., \boldsymbol{x}_{m-1}]; \quad \boldsymbol{a} = [a, a, ..., a]^{\top}, \tag{7}$$

and the matrix $D_m^{(1)} \in \mathbf{R}^{m\times m}$ being

$$D_m^{(1)} = \begin{bmatrix} 0 & -\theta_{01} & 0 & & 0 \\ 0 & +\theta_{01} & -\theta_{12} & & 0 \\ 0 & 0 & +\theta_{12} & \ddots & 0 \\ \vdots & & & \ddots & -\theta_{m-2,m-1} \\ 0 & 0 & 0 & & +\theta_{m-2,m-1} \end{bmatrix}, \tag{8}$$

where $\theta_{k-1,k} = 1/(t_k - t_{k-1}) > 0$. Equation 6 shows itself as a special case of the generic matrix equation

$$XD = AX, \tag{9}$$

where D is a matrix acting on the *temporal* indices of the data matrix X, while A is a matrix acting on the *spatial* indices of X. We will return to this point later in this section.

1.4 DMD with control

Among the extensions of the DMD, the DMD with control (DMDc) [9] is a method for analysis of time-series data corresponding to a non-autonomous dynamical system. The term *non-autonomous* means the existence of the exogenous external forcing, whose time-dependence is not affected by the status of the dynamical system itself. One example is the forced oscillation, in which one apply the external forcing to the oscillator. The generic form of the non-autonomous dynamical system of our interest is expressed as follows:

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = A\boldsymbol{x}(t) + B\boldsymbol{u}(t), \tag{10}$$

with $\boldsymbol{x}$ being the n-dimensional state vector and $\boldsymbol{u}$ being the ℓ-dimensional external forcing vector. The coefficients $A \in \mathbf{R}^{n\times n}$ and $B \in \mathbf{R}^{n\times \ell}$ are constants in time. The DMDc is formulated as the estimation of matrices A, B from the observations X and the external forcing $\Upsilon = [\boldsymbol{u}_0, \boldsymbol{u}_1, ..., \boldsymbol{u}_{m-1}]$ as follows:

$$\begin{bmatrix} A & B \end{bmatrix} \simeq \begin{bmatrix} \bar{A} & \bar{B} \end{bmatrix} = XD_m^{(1)} \begin{bmatrix} X_{1:|0} \\ \Upsilon_{1:|0} \end{bmatrix}^{+}, \tag{11}$$

where $[A\ B]$ is a block matrix with two blocks A and B. The application of the standard procedure of singular-value decomposition (SVD) leads to the reduced representation [9].

1.5 Other Related Works

Since its highly extensive algorithmic structure, the DMD is also extended by many researchers. The residual DMD (resDMD) [10] is an neural network-based method with ResNet-like blocks with its reference is replaced by the results of the DMD algorithm. The incorporation of neural nets enables resDMD to deal with time-series data that is best fitted by a highly nonlinear time-evolution models. The optimized DMD (optDMD) [1] and bagging, optimized DMD (BOP-DMD) [15] are extensions of DMD, with a nonlinear optimization gives a better de-biasing while a bagging of snapshots gives a better convergence to the optimization and enables an uncertainty quantification.

1.6 Memory Effects

Despite the huge success of the abovementioned methods, there is a room for improvements in the way DMD incorporates the time-evolution models. Existing DMDs are either using the first-order ODE (exponential) model [23, 9, 2] or, using neural network-based method to deal with the nonlinearity in the time-evolution models [10]. As another extension of DMD, it is possible to extend the time-evolution model within the linear model but with the memory effects.

Many known fundamental physical processes are governed by the first- or second-order ODE, leading to exponential-like behaviors in time. Thus the exponential time-evolution plays a crucial role in various realms in physics.

Although the microscopic, fundamental physics are governed by low-order ODEs (i.e., the memoryless equations of motion), systems with strong coupling with an external system, or a *reservoir*, behave differently. Suppose we know the internal state of the system, and we have no information on the internal state of the reservoir. Once we apply a stimulus to the system, it causes a change of the system state, leading to a change of the reservoir state. Subsequently, the change in the reservoir might also affects the system itself via the system-reservoir surface. This sort of *indirect* effects via an external system leads to a memory effect, in which the time-evolution of the system seems to be affected not only by the *current* status, but also by the *history* of its time-evolution [20, 3], leading to an equation of motion described by fractional differential equations (e.g., [19]).

As explained above, the *memory* of a system means a nonlocal behavior in time-domain, mostly arising from limitations of our observations.

1.7 Aim of This Paper

In this paper, we propose a DMD-based method, DMD with *memory* (DMDm), whose time-evolution models are described by a wider class of equations that allows the descriptions of the system with memory effects. We restrict our attention to the definition of exact DMD given by Tu *et al.* (see Definition 1 of [23]). Let us assume a linear functional relation $y = \pi(x)$ for two time-series data $x : t \mapsto x(t)$ and $y : t \mapsto y(t)$. If the value $y(t)$ is *not* affected by the values $x(t')$ for $t' > t$, the linear map π is said to be *causal*, since the operation is performed without knowing the *future* values of x.

It is worth noting that the construction of an eigenfunction of a causal linear operator can also be formulated as an initial-value problem. This is also true in the discrete-time time-series data and the discretized representation of the linear operator D_π. The discretized representation of the eigenfunction $z : t \mapsto z(t)$ corresponding to an eigenvalue λ is the array of numbers $\big[z(t_0), z(t_1), ..., z(t_{m-1})\big]$, that satisfies the matrix equation:

$$\big[z(t_0), z(t_1), ..., z(t_{m-1})\big] D_\pi = \lambda \big[z(t_0), z(t_1), ..., z(t_{m-1})\big], \quad z(t_0) = z_0. \tag{12}$$

For a causal D_π, the element of the array $z(t_k)$ is constructed by an initial-value problem, or, by applying the *transition operator* $\mathcal{K}^{t_k}$ to the initial status of the system. The above discussion leads to the mode decomposition of the form, similar to Equation 5:

$$\boldsymbol{x}(t) = \sum_{j=0}^{r-1} \vec{\phi}_j F_{\pi,\lambda_j,1}(t) b_j; \quad b_j = (\boldsymbol{x}_0)_j. \tag{13}$$

In Equation 13, the vector $\vec{\phi}_j \in \mathbf{R}^n$ is the DMD mode of the problem, and the function F_{π,λ,z_0} is the solution of the initial value problem $\pi(z)(t) = \lambda z(t)$ $(z(0) = z_0)$. Note that the eigenvector in spatial direction $\vec{\phi}_j$ and the eigenfunction in time domain $F_{\pi,\lambda_j,1}$ share the common eigenvalue λ_j, corresponding to the factorization of the solution into temporal and spatial parts. Also it is worth noting that this proposed framework contains the exact DMD, because

$$F_{\frac{\mathrm{d}}{\mathrm{d}t},\lambda,1} = \exp(\lambda t). \tag{14}$$

As an solid example of the above discussions, we use the fractional derivative [3, 20, 21] for the time-domain transformation. Fractional derivative of real-valued [3, 21] and complex-valued orders [13] are quite useful in physics with memory effects with power-law [20, 3]. The idea of introducing fractional (non-integer) order derivative is also attracting attention in the control theory [22, 8], with famous $\mathrm{PI}^\lambda\mathrm{D}^\mu$-controllers [8]. The α-th order fractional integral of an integrable function f and a real value $\alpha > 0$ is given by the following [21].

$$(I^\alpha f)(t) = \frac{1}{\Gamma(\alpha)} \int_{-\infty}^{t} \mathrm{d}t' \, \frac{f(t')}{(t-t')^{1-\alpha}}. \tag{15}$$

The fractional integral satisfies the following properties for any integrable functions f, g and real values $\alpha, \beta > 0$:

1. $I^\alpha(f+g) = I^\alpha f + I^\alpha g$,
2. $I^\alpha I^\beta f = I^{\alpha+\beta} f$,
3. $I^1 f(t) = \int_{-\infty}^{t} \mathrm{d}t' \, f(t')$.

The first and second conditions correspond to linearity with respect to the function and additivity for the order, respectively. The third condition is the equivalence of I^α with the Riemann integral for the case $\alpha = 1$. Noting that the integral is the inverse operation of the derivative, one can construct α-th order derivative fot any $\alpha \in \mathbf{R}$ [21] for an integrable function $f : (-\infty, t_1] \to \mathbf{R}$ $(t_1 > -\infty$ is the upper bound of the domain of $f)$.

In the following text, we will see a concrete definition for the α-th order fractional derivative $\frac{\mathrm{d}^\alpha f}{\mathrm{d}t^\alpha}$ for an integrable function $f : \mathbf{R} \supset [t_0, t_1] \to \mathbf{R}$ and show the idea of *fractional* DMD, or fracDMD, in which fractional derivative determines the time-dependency of each DMD modes, instead of the first-order derivative.

The remainder of the paper is organized as follows: in Section 2, we investigate the property of the causal linear functional and its eigenfunctions. In section 3, we introduce DMDm, and in Sections 4-6 we introduce fracDMD as an example of DMDm. In Sections 7, 8, we apply fracDMD to synthetic time-series data. Section 9 is devoted for the discussions of the numerical results.

Throughout this paper, we denote any complex matrix by $\bullet$, and any scaler by $-$. The column vector $\bullet_k$ and slice $\bullet_{k:\ell}$ of a matrix for $0 \le k \le \ell \le m$ are defined as, for a matrix $M = \left[\mathbf{m}_0, \mathbf{m}_1, ..., \mathbf{m}_{m-1}\right] \in \mathbf{C}^{n\times m}$,

$$M_k = \mathbf{m}_k, \quad M_{k:\ell} = [\mathbf{m}_k, ..., \mathbf{m}_{\ell-1}]. \tag{16}$$

Also, we use $M_{k:} = M_{k:m}$ and $M_{:\ell} = M_{0:\ell}$ for short. A block matrix with blocks A_i $(i = 1, 2, ..., q)$ being stacked column-wise is denoted by $\left[A_1; A_2; ...; A_q\right]$, i.e.,

$$\left[A_1; A_2; ...; A_q\right] = \begin{bmatrix} A_1 \\ A_2 \\ ... \\ A_q \end{bmatrix}, \tag{17}$$

for short.

2 Time-Evolutions as Eigenvalue Problems

In this section, we investigate the conditions for the linear operator π introduced in the previous section. Let us denote the set of one-dimensional time-series data by $\mathbf{C}^{\mathcal{T}}$. A time-series datum $V \in \mathbf{C}^{\mathcal{T}}$ is a map from set of time points $\mathcal{T}$ to complex numbers, i.e., $V : \mathcal{T} \to \mathbf{C}$. Let us denote the number of time points by $m = |\mathcal{T}| > 0$. In the followings, we assume that $m < \infty$.

2.1 Causal Linear Operator

Let us define a *causal linear operator* $\pi : \mathbf{C}^{\mathcal{T}} \ni V \mapsto W \in \mathbf{C}^{\mathcal{T}}$ as a linear functional $\pi \in \hom(\mathbf{C}^{\mathcal{T}}, \mathbf{C}^{\mathcal{T}})$ between two time-series that satisfies the following condition.

- π is causal: $\pi(V)_{:k} = \pi(V_{:\ell})_{:k}$ for any ℓ, k satisfying $1 \leq k \leq \ell \leq |\mathcal{T}|$.

The *causality* defined above corresponds to the fact that the i-th component W_i of the time-series $W = \phi(V)$ is not dependent on V_k for $k > i$.

2.2 Eigenfunction of Causal Linear Operator

For a given causal linear operator $\pi \in \hom(\mathbf{C}^{\mathcal{T}}, \mathbf{C}^{\mathcal{T}})$ and an initial value V_0, we can construct the *eigenfunction* of π by an iterative method as follows. First, we introduce the (row) vector representation of V as $V = \left[V_0, V_1, ..., V_{m-1}\right]$ with $V_k \in \mathbf{C}$ $(k = 0, 1, ..., m-1)$. The matrix representation of π is then introduced by the followings:

$$VD_\pi = \pi(V), \tag{18}$$

where, in the left-hand side, a standard matrix multiplication is assumed. Note that, due to the causality of π, the matrix representation D_π is an upper-triangle matrix:

$$D_\pi = \begin{bmatrix} \pi_{00} & * & * & \cdots & * \\ 0 & \pi_{11} & * & \cdots & * \\ \vdots & & \ddots & & \\ 0 & & & \pi_{m-2,m-2} & * \\ 0 & 0 & \cdots & 0 & \pi_{m-1,m-1} \end{bmatrix}. \tag{19}$$

The eigenvalue problem for the linear operator π is expressed as the following eigenvalue equation for the row vector V and the matrix D_π:

$$VD_\pi = \lambda V. \tag{20}$$

Since the inverse of an upper-tirangle matrix is also an upper-triangle matrix, the above equation becomes,

$$V = \lambda V D_\pi^{-1}, \tag{21}$$

with

$$D_\pi^{-1} = \begin{bmatrix} (\pi_{00})^{-1} & * & * & \cdots & * \\ 0 & (\pi_{11})^{-1} & * & \cdots & * \\ \vdots & & \ddots & & \\ 0 & & & (\pi_{m-2,m-2})^{-1} & * \\ 0 & 0 & \cdots & 0 & (\pi_{m-1,m-1})^{-1} \end{bmatrix}. \tag{22}$$

Here we assumed that D_π is invertible. Thus the eigenfunction of π is given by the following successive calculations for given V_1:

$$V_k = \left[\frac{\lambda V_{:k|0}(D_\pi^{-1})_\times}{1 - \pi_{kk}^{-1}\lambda}\right]_k \quad (k = 2, 3, ...), \tag{23}$$

where, the zero-padded vector $\bullet_{:k|0}$ and the off-diagonal parts of a matrix $\bullet_{\times}$ is defined as the followings:

$$V_{:i|0} = \big[V_0, V_1, ..., V_{i-1}, \overbrace{0, ..., 0}^{m-i}\big], \quad (a_{ij})_{\times} = \begin{cases} a_{ij} & (i \neq j) \\ 0 & (i = j). \end{cases} \tag{24}$$

The procedure to obtain the eigenmode of π from a given initial value V_0 is summarized in 1.

Algorithm 1 Values of the Eigenfunction $V_i = F_{\pi,\lambda,V_0}(t_i)$ for a Causal Linear Operator π

Require: causal linear operator π, real-valued parameter λ, initial value $V_0 \in \mathbf{R}$, time $t_i \in \mathcal{T}$
  $V \leftarrow [V_0, 0, ..., 0] \in \mathbf{C}^m$
  for $k \in \{1, ..., m-1\}$ **do**
    $V_k \leftarrow \left[\frac{\lambda V(D_\pi^{-1})_{\times}}{1-\pi_{kk}\lambda}\right]_k$
    $V \leftarrow \big[V_0, V_1, ..., V_k, 0, ..., 0\big] \in \mathbf{C}^m$
  end for
  return V

Using a sequence of time-points $\mathcal{T}$ with its maximum time interval $\Delta t_{\max} \to +0$, 1 can approximate a smooth function.

3 Dynamic Mode Decomposition with Memory

Using the causal linear operators and their eigenfunctions introduced in the previous section, we now postulate the idea of DMD with memory (DMDm). DMDm is a new method to include the memory effects in DMD framework. Itemploys a causal linear operator instead of the difference operator used in the existing DMD. This gives us the ability to handle the effects from the past data in the time-evolution model without losing advantages of the DMD framework.

3.1 Definition of the Model

Assume we have a causal linear functional π. We introduce a linear time-evolution model of the form,

$$\pi(\boldsymbol{x})(t) = A\boldsymbol{x}(t), \tag{25}$$

where $A \in \mathbf{R}^{n\times n}$ is a constant matrix. Restricting t onto the (finite) set of time points $\mathcal{T}$, we can discretize this continuous-time time-evolution model to get the matrix representation

$$XD_\pi = AX, \tag{26}$$

where we have a matrix A and n-dimensional time-series data X, instead of a scaler λ and 1-dimensional time-series V in Equation 20, respectively. Given the SVD $X = U\Sigma V^*$ with unitary matrices U and V, we can transform the Equation 26 as follows:

$$U^*XD_\pi = U^*AU(U^*X). \tag{27}$$

For a diagonalizable matrix $A \sim \mathrm{diag}(\lambda_0, ..., \lambda_{n-1})$, we can always find an array of new variables $\Xi = \mathcal{U}^{-1}U^*X$ (i.e., we define a new variable by $\boldsymbol{\xi}(t) = \mathcal{U}^{-1}U^*\boldsymbol{x}(t)$ and $\Xi = \big[\boldsymbol{\xi}_0, \boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_{m-1}\big]$) for an appropriate matrix $\mathcal{U}$ such that:

$$\Xi D_\pi = \mathrm{diag}(\lambda_0, ..., \lambda_{n-1})\Xi. \tag{28}$$

The above equation leads to the solution for the original variable $\boldsymbol{x}$ of the following form

$$\boldsymbol{x}(t) = U\mathcal{U}\begin{bmatrix} F_{\pi,\lambda_0,\xi_0(0)}(t) \\ \vdots \\ F_{\pi,\lambda_{n-1},\xi_{n-1}(0)}(t) \end{bmatrix} = \sum_{i=0}^{n-1} (U\mathcal{U}\mathbf{e}_i)F_{\pi,\lambda_i,1}(t)\xi_i(0), \tag{29}$$

where the function F is defined in 1 and $\mathbf{e}_i = \left[\delta_{0i}, \delta_{1i}, ..., \delta_{m-1,i}\right]^\top$ with the unit tensor δ_{ij}. Note that the above expression is similar to Equation 13.

3.2 Model Fitting via DMD Scheme

The goal is to find an appropriate matrix A with which one can explain the data X on time points $\mathcal{T}$. We assume the model Equation 26 for the observed data X. The best fit parameter A that achieves a least-square for $\pi(x)(t)$ is,

$$A = \arg\min_{A'} \|XD_\pi - A'X\|_F = XD_\pi X^+. \tag{30}$$

Similar to Equation 3, we can compute A using the SVD:

$$A = XD_\pi V\Sigma^{-1}U^*. \tag{31}$$

The low-rank approximation can also be performed in a similar way as in Equation 4.

3.3 DMDc with Memory

The DMDc scheme is readily applicable for the proposed method. The resultant method, DMD with control and memory (DMDcm) assumes the following time-evolution model.

$$\pi(\boldsymbol{x})(t) = A\boldsymbol{x}(t) + B\boldsymbol{u}(t), \tag{32}$$

which corresponds to the matrix representation

$$XD_\pi = AX + B\Upsilon, \tag{33}$$

where $A \in \mathbf{C}^{n\times n}$ and $B \in \mathbf{C}^{n\times \ell}$ are coefficient matrices. The DMDc prescription gives the direct calculation of the optimal coefficients, similar to Equation 11

$$\begin{bmatrix} A & B \end{bmatrix} = XD_\pi \begin{bmatrix} X \\ \Upsilon \end{bmatrix}^+. \tag{34}$$

1 is readily applicable for multi-dimensional case, and the time-evolution of the model Equation 33 for given initial value $\boldsymbol{x}_0$ and external forcing Υ is obtained by the following successive calculations:

$$\boldsymbol{x}(t_i) = \left[\frac{(AX_{:i|0} + B\Upsilon_{:i|0})(D_\pi^{-1})_\times}{\hat{1}_n - \pi_{ii}A}\right]_i \quad (i = 1, 2, ...). \tag{35}$$

The coefficients A, B can be estimated by the following equation:

$$\begin{bmatrix} A & B \end{bmatrix} \simeq \begin{bmatrix} \bar{A} & \bar{B} \end{bmatrix} = XD_\pi \begin{bmatrix} X \\ \Upsilon \end{bmatrix}^+. \tag{36}$$

As is easily seen, this is a natural extension of DMDc, justifying the name DMDcm.

4 Integer-order Dynamic Mode Decomposition

The finite difference method is employed for the first-order differential equation in Equation 10. A similar procedure can be used for a-th order differential equation for $a = 2, 3, 4, ...$ as well. The resultant method is the arbitrary integer-order DMD (intDMD), as described below.

$$A = X\Delta_m^{(a)} X^+_{:m-\ell+1}, \tag{37}$$

with the matrix representation of the ℓ-th order derivative $\Delta_m^{(\ell)} \in \mathbf{R}^{m\times(m-\ell)}$. Although the extension to a positive-integer order seems to be nontrivial, the same effect is actually achieved by introducing auxiliary variables $X^{(n)} = X\Delta_m^{(n)}$ for $n = 1, 2, ..., a-1$ and using the extended matrix $\mathbf{X} = [X; X^{(1)}; ...; X^{(a-1)}]$ in the DMD framework.

5 Fractional Dynamic Mode Decomposition

The order of the differential operator can be extended to any real-valued numbers [21]. The term for this generalized differentiation, *fractional derivative* is somehow misleading, since we can specify an irrational number $\alpha \in \mathbf{R}/\mathbf{Q}$ as the order of the differentiation. One of the definition of the α-th order differential of a smooth integrable function $f : \mathbf{R} \to \mathbf{R}$ is the Caputo derivative, as shown in the followings:

$$\frac{\mathrm{d}^\alpha f}{\mathrm{d}t^\alpha} = \frac{1}{\Gamma(\nu)} \int_0^t \mathrm{d}t' \, \frac{f^{(\ell)}(t')}{(t-t')^{1-\nu}} \quad (t > 0,\ \nu = \ell - \alpha,\ \ell = \max(0, \lceil \alpha \rceil)), \tag{38}$$

where $f^{(\ell)}$ is the ℓ-th order derivative of the function f and the ceiling $y = \lceil x \rceil$ is the minimum integer $y \in \mathbf{Z}$ such that $y - x > 0$. Note that the right-hand side of the definition above corresponds to the case $\alpha = \nu$ in the definition of the fractional *integral* in Equation (15) with $0 < \nu \leq 1$, and thus we can circumvent a divergence. In this section, we consider real-valued functions for simplicity.

One can construct the corresponding *eigenmode* for the α-th order derivative with an initial value. To see this, let us denote the discrete representation of a generic-order fractional differential operator $\frac{\mathrm{d}^\alpha}{\mathrm{d}t^\alpha}$ by $\mathcal{D}_m^{(\alpha)}$ for $\alpha \in \mathbf{R}$. We first define the discrete representation of ℓ-th order integer-order derivative $D_m^{(\ell)}$ for $\ell \in \mathbf{Z}_+$ (e.g., we can define $D_m^{(\ell)} = \left[O_{m,\ell} \ \Delta_m^{(\ell)} \right]$ with $O_{a,b}$ being the $a \times b$ zero matrix). The explicit expression of $\mathcal{D}_m^{(\alpha)}$ is calculated using the Caputo fractional differential [21] as follows,

$$\mathcal{D}_m^{(\alpha)} = D_m^{(\ell)} \left[\boldsymbol{w}_0^{(\ell-\alpha)}, \boldsymbol{w}_1^{(\ell-\alpha)}, ..., \boldsymbol{w}_{m-1}^{(\ell-\alpha)} \right], \quad \ell = \max(0, \lceil \alpha \rceil). \tag{39}$$

At the lowest order, the *weight* $\boldsymbol{w}_k^{(\nu)}$ is approximated as follows:

$$(\boldsymbol{w}_k^{(\nu)})_i = \begin{cases} \frac{1}{\Gamma(\nu)} \frac{1}{(t_k - t_i)^{\nu+1}} & (i < k) \\ 0 & (i \geq k) \end{cases}. \tag{40}$$

Thus the derivative above is decomposed into the ℓ-th order (integer-order) differential and the ν-th order fractional differential (i.e., fractional integral of order $0 < \nu \leq 1$). Note, by construction, that $\nu > 0$, and hence the weight $(\boldsymbol{w}_k^{(\nu)})_i \to 0$ for $|k - i| \to \infty$. Clearly, the discretized representation $\mathcal{D}^{(\alpha)}$ satisfies the linearity and the causality, ensuring the existence of eigenmodes. In an actual numerical analysis, one might use a more elaborate implementations with higher-order schemes. In the continious limit, the eigenfunction of the fractional derivative operator is known to be expressed by the Mittag-Leffler functions [4, 5].

For a given time-series data $X \in \mathbf{R}^{n \times m}$ arranged in a matrix form and a fixed order of derivative α, the best fit matrix A for the model equation $X\mathcal{D}^{(\alpha)} = AX$ is obtained as follows:

$$A = X\mathcal{D}^{(\alpha)} X^+. \tag{41}$$

The above expression is analogous to Equation 3 for the first-order (ordinary) DMD. The SVD of the matrix $X = \hat{U}\hat{\Sigma}\hat{V}^* \simeq \hat{U}_r \hat{\Sigma}_r \hat{V}_r^*$ with rank $0 < r \equiv \mathrm{rank}(\hat{\Sigma}_r) < n$ and unitary matrices $\hat{U}, \hat{U}_r, \hat{V}, \hat{V}_r$, diagonal matrices $\hat{\Sigma}, \hat{\Sigma}_r$ yields a low-rank representation of the dynamics for $\boldsymbol{\xi} = \hat{U}_r^* \boldsymbol{x} \in \mathbf{R}^r$,

$$\frac{\mathrm{d}^\alpha \boldsymbol{\xi}}{\mathrm{d}t^\alpha} = \Lambda \boldsymbol{\xi}, \quad \Lambda = \hat{U}_r^* X \mathcal{D}^{(\alpha)} \hat{V}_r \hat{\Sigma}_r^{-1}. \tag{42}$$

If the order α is unknown, the matrix A and the fractional order α is now estimated by 2.

Algorithm 2 Grid search for order α

Require: data in matrix form X, candidates $\boldsymbol{\alpha} = \{\alpha_0, \alpha_1, ...\}$
  if $\alpha \in \boldsymbol{\alpha}$ **then**
    $A \leftarrow X\mathcal{D}^{(\alpha)}X^+$
    $L(\alpha) \leftarrow \|X\mathcal{D}^{(\alpha)} - AX\|_F$
  end if
  $\alpha_* \leftarrow \arg\min_\alpha (L(\alpha))$

6 Fractional Dynamic Mode Decomposition with Control

The original DMD formulation is only for autonomous systems, i.e., it does not include the effects from the environment or an outer controlling system. As we have seen in the previous sections, DMDc [9] is an extension of the DMD for the non-autonomous systems with exogenous terms. The DMDc can further be extended to the dynamical systems which obey the fractional-order differential equations. Suppose we have a dynamical system

$$\frac{\mathrm{d}^\alpha \boldsymbol{x}}{\mathrm{d}t^\alpha} = A\boldsymbol{x} + B\boldsymbol{u}(t), \tag{43}$$

with the state vector $\boldsymbol{x} \in \mathbf{R}^n$ and the external forcing (exogenous input) vector $\boldsymbol{u}(t) \in \mathbf{R}^\ell$. We construct matrices $X = [\boldsymbol{x}_0, \boldsymbol{x}_1, ..., \boldsymbol{x}_{m-1}]$ and $\Upsilon = [\boldsymbol{u}_0, \boldsymbol{u}_1, ..., \boldsymbol{u}_{m-1}]$. In an analogy to Equation 11, we obtain the following estimation for the coefficients.

$$\begin{bmatrix} A & B \end{bmatrix} \simeq \begin{bmatrix} \bar{A} & \bar{B} \end{bmatrix} = X\mathcal{D}^{(\alpha)} \begin{bmatrix} X \\ \Upsilon \end{bmatrix}^+. \tag{44}$$

We can also make use of the SVDs, $X = \hat{U}\hat{\Sigma}\hat{V}^* \simeq \hat{U}_r\hat{\Sigma}_r\hat{V}_r^*$ and $[X; \Upsilon] = \tilde{U}\tilde{\Sigma}\tilde{V}^* \simeq \tilde{U}_p\tilde{\Sigma}_p\tilde{V}_p^*$ with $p > r$, to get an approximated low-rank representation of the dynamics for $\boldsymbol{\xi} = \hat{U}^*\boldsymbol{x} \in \mathbf{R}^r$, as follows:

$$\frac{\mathrm{d}^\alpha \boldsymbol{\xi}}{\mathrm{d}t^\alpha} = \Lambda\boldsymbol{\xi} + \Gamma\boldsymbol{u}, \quad \begin{bmatrix} \Lambda & \Gamma \end{bmatrix} \simeq \hat{U}_r^* X\mathcal{D}^{(\alpha)}\tilde{V}_p\tilde{\Sigma}_p^{-1}\tilde{U}_p^* \begin{bmatrix} \hat{U}_r^* & 0 \\ 0 & \hat{1}_{p-r} \end{bmatrix}. \tag{45}$$

Thus we can analyze a time-series data of input $\boldsymbol{u}(t)$ and the output $\boldsymbol{x}(t)$ for a system in the fracDMD framework, as well as for autonomous systems. The fracDMD algorithm is explicitly shown in 3. The function $\texttt{SVD}(-, r)$ is the SVD of a matrix with rank r.

Algorithm 3 fracDMD

Require: input data in matrix form Υ, output data in matrix form X, ranks $p > r > 0$, α
  $\hat{U}_r, \hat{\Sigma}_r, \hat{V}_r \leftarrow \texttt{SVD}(X, r)$
  $\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p \leftarrow \texttt{SVD}(\begin{bmatrix} X; \Upsilon \end{bmatrix}, p)$
  $X' \leftarrow X\mathcal{D}^{(\alpha)}$
  $\begin{bmatrix} \Lambda & \Gamma \end{bmatrix} \leftarrow \hat{U}_r^* X'\tilde{V}_p\tilde{\Sigma}_p^{-1}\tilde{U}_p^* \begin{bmatrix} \hat{U}_r^* & 0 \\ 0 & \hat{1}_{p-r} \end{bmatrix}$

7 Quantitative Evaluations for the Models

The fracDMD, as well as the existing DMD, is to minimize the Frobenius norm of the matirix $\|X' - AX - B\Upsilon\|$, with X' being the fractional derivative $X\mathcal{D}^{(\alpha)}$ for fracDMD. The Frobenius norm-based reconstruction error $\mathcal{L}^{(\alpha)}_{\text{Frobenius}}(A, B; X) = \|X\mathcal{D}^{(\alpha)} - AX - B\Upsilon\|$ is considered to be the sum of squared

reconstruction error in α-th order time derivative, with $AX + B\Upsilon$ regarded as the reconstruction by the model.

Another way to evaluate the model is to use the sum of squared errors (SSE) in the reconstructed states x_i $(i = 0, 1, ..., m-1)$. The explicit expression of SSE $\mathcal{L}^{(\alpha)}_{\mathrm{SSE}}(A, B; X)$ as a function of α, A, B, and $X = [\boldsymbol{x}_0, \boldsymbol{x}_1, ..., \boldsymbol{x}_{m-1}]$ is given by,

$$\mathcal{L}^{(\alpha)}_{\mathrm{SSE}}(A, B; X) = \sum_{i=0}^{m-1} |\boldsymbol{x}(t_i) - \boldsymbol{x}_i|^2, \tag{46}$$

with $\boldsymbol{x}(t_i)$ being the time-evolution with coefficients A, B from the initial value $\boldsymbol{x}_0$ for the fractional differential Equation 43, whose explicit expression is shown in Equation (29).

8 Numerical Experiments

We perform numerical experiments to show the utility of fracDMD against the synthetic data. We numerically generate the solution of the fractional oscillator [18] analyzed by Svenkeson *et al.* [20] in a context of spectral decomposition. Svenkeson *et al.* performed numerical tests on the real-time behavior of a single, noise-free fractional oscillator with known parameters to show that the fractional-order calculus is useful in analyzing the memory effects. We extend their method to include a multi-dimensional, noisy fractional oscillators. We also added an observation matrix, so that the mode reconstruction becomes highly nontrivial.

8.1 Numerical Set-up

The one-dimensional fractional linear oscillator is given by the following equation of motion [20]:

$$\frac{\mathrm{d}^\nu \boldsymbol{y}_a}{\mathrm{d}t^\nu} = Q_a \boldsymbol{y}_a; \quad \boldsymbol{y}_a = \begin{bmatrix} x_a \\ v_a \end{bmatrix}, \quad Q_a = \begin{bmatrix} 0 & 1 \\ -\omega_a^2 & 0 \end{bmatrix}, \tag{47}$$

where the frequency $\omega_a > 0$ is a real-value parameter and $a = 0, 1, ...$ is the index. Let us further consider a multi-dimensional time-evolution equation, as follows:

$$\frac{\mathrm{d}^\nu \boldsymbol{x}}{\mathrm{d}t^\nu} = \Phi \boldsymbol{x}; \quad \boldsymbol{x} = \begin{bmatrix} \boldsymbol{y}_0 \\ \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_{k-1} \end{bmatrix} \in \mathbf{R}^{2k}, \quad \Phi = \begin{bmatrix} Q_0 & & & \\ & Q_1 & & \\ & & \ddots & \\ & & & Q_{k-1} \end{bmatrix} \in \mathbf{R}^{2k \times 2k}, \tag{48}$$

where $\boldsymbol{y}_a$ and Q_a denote the state vector in the single-particle phase space and the 2×2 matrix, respectively. We also consider the observation equation as follows:

$$\boldsymbol{z}(t) = R\boldsymbol{x}(t) + \boldsymbol{\epsilon}(t), \quad \epsilon_i(t) \overset{\mathrm{iid}}{\sim} \mathcal{N}(0, \sigma^2) \quad (i = 0, 1, ..., 2k-1), \tag{49}$$

where $R \in \mathbf{R}^{n \times 2k}$ is the constant observation matrix, each element of $\boldsymbol{\epsilon}(t) \in \mathbf{R}^n$ is the independent and identically distributed (iid) Gaussian noise and $\boldsymbol{z} \in \mathbf{R}^n$ represents the observed signal. This numerical set-up is useful to describe the situation in which the oscillators do not interact to each other but the resulting signal is the superposition of oscillators.

Hereafter we assume that $n \geq 2k$. The observation matrix R consists of the randomly sampled $2k$ basis vectors $v_{s(\mu)} \in \mathbf{R}^n$ $(\mu = 0, 1, ..., 2k-1)$ for a random orthonormal basis $\{v_0, v_1 ..., v_{n-1}\}$ and a permutation $s \in \mathfrak{S}_n$, as follows:

$$R = \left[v_{s(0)}, v_{s(1)}, ..., v_{s(2k-1)}\right] \in \mathbf{R}^{n \times 2k}. \tag{50}$$

Note that $s(i) \neq s(j)$ for $i \neq j$.

8.2 Numerical Tests

We performthe numerical tests to determine the frequency ω_a $(a = 0, 1, ..., k-1)$ in Equations (47), (48). The eigenvalues of the coefficient matrix Φ is $\pm\sqrt{-1}\omega_a$ $(a = 0, 1, ..., k-1)$. We also denote the eigenvalues of the coefficient matrix A obtained by fracDMD (without control) by λ_i $(i = 0, 1, ..., 2k-1)$. Let us assume that $\omega_0 > \omega_1 > ... > \omega_{k-1} > 0$ and $\mathrm{Im}\,\lambda_0 > \mathrm{Im}\,\lambda_1 > ... > \mathrm{Im}\,\lambda_{k-1} > \mathrm{Im}\,\lambda_{2k-1} > \mathrm{Im}\,\lambda_{2k-2} > ... > \mathrm{Im}\,\lambda_k$ for simplicity.

The error L of the frequency estimation is now given by the followings:

$$L = \sum_{i=0}^{k-1} |\lambda_i - \omega_i|^2 + \sum_{i=0}^{k-1} |\lambda_{i+k} + \omega_i|^2. \tag{51}$$

In the full reconstruction case, $\lambda_i = \sqrt{-1}\omega_i$ and $\lambda_{k+i} = -\sqrt{-1}\omega_i$ $(i = 0, 1, ..., k-1)$, so that $L = 0$.

We show the frequency reconstruction error L for various values of noise standard deviation σ and the dimension of the observation n in 8.2 below. We assume that the rank $2k$ and the order ν of the system Equation (48) are known. We use the second-order numerical discretization of the Caputo fractional differential, instead of the first-order scheme shown in Equations (39), (40). We also modified Equation (41) so that we can use the fractional *integral* instead of the fractional differential with positive order α in order to achieve a better numerical convergence.

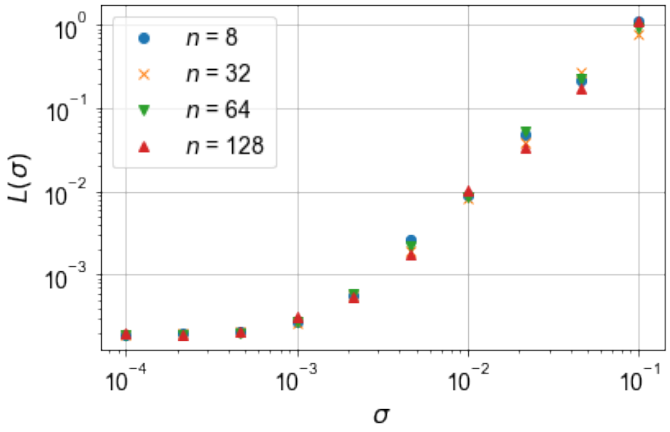
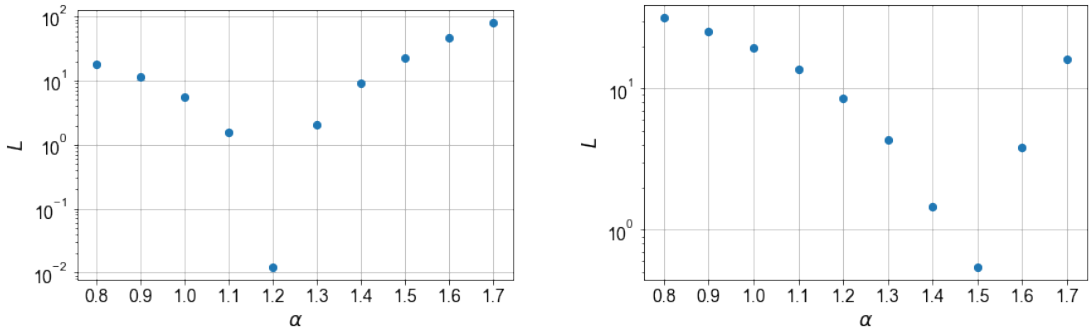


Figure 1: The frequency reconstruction error L against the noise standard deviation σ, for various values of observation size n. The numerical set-ups are as follows: $k = 4$, $\omega_i = i + 1$ $(i = 0, 1, 2, 3)$, and the order of the system euqation Equation (48) is set to $\nu = 1.2$. The fracDMD parameters are as follows: the SVD rank r is set to the actual system size $2k$, and the order α is set to the actual value ν. Each mark denotes a mean of 10 numerical calculations with different noise realizations and initial conditions. The initial conditions are randomly chosen so that the initial energy of each oscillator is unity.

We can see that the reconstruction error L is an increasing function of σ for $\sigma > 10^{-3}$, while L is almost constant for $\sigma < 10^{-3}$. This implies that the major part of the reconstruction error vanishes with $\sigma \to 0$, leaving the constant part of the error for small σ.



(a) $\nu = 1.2$ (b) $\nu = 1.5$

Figure 2: The frequency reconstruction error L against the order α used in the fracDMD. (a) $\nu = 1.2$, (b) $\nu = 1.5$. In both tests, $\sigma = 10^{-2}$ and $n = 8$. The other numerical set-ups are the same as in 8.2. The fracDMD parameters are as follows: the SVD rank r is set to the actual system size $2k$.

As we can see, the function $L(\alpha)$ has with its minima located around $\alpha = \nu$, indicating that a *wrong* value of α leads to a larger value of reconstruction error.

9 Discussion

The fractional oscillator is a model for the oscillator with power-law memory effects. The reconstruction of the frequencies in the previous section is in a good agreement with the ground truth. The reconstruction error increases with the increasing noise standard deviations, while the dimension of the observation space does not affect the error severely. We thus conclude that the fracDMD can reconstruct the isolated fractional oscillators well.

We also performed numerical experiments for various values of α. It is shown that the reconstruction error L has its minima around $\alpha = \nu$. This implies that our method is also useful to estimate the order of the system equation. Noting that the case $\alpha = 1$ roughly corresponds to the DMD, we can also infer that our method achieves a better reconstruction than the DMD for the fractional oscillators. Although this inference seems exact, the introduction of a higher-order scheme and the fractional integral might lead to a discrepancy between DMD and fracDMD results even for $\alpha = 1$ cases.

Our method can be used to model unknown physical processes e.g., mechanical motion of industrial machines and thermal systems. One possible way to use the method is to estimate the memory effect in the system. If the reconstruction error by the fracDMD has its minima around $\alpha = 1$, one may conclude that the system is memoryless; otherwise, one can use the optimal value of α to include the memory effects to the model.

Among possible extensions, nonlinear coupled oscillators are one possible way to extend our analysis. Another way to include more complex situations is to use the non power-law memories. In our current numerical set-up, we use power-law memories, which is mathematically shown to be equivalent to the fractional-order equations of motion [20]. However, the memories in real data can, in principle, decay according to any smooth functions.

In the reconstruction considered in the previous section, we assume that we know the actual size of the problem (i.e., the system size). This assumption makes the problem easy to approach. In a realistic situation, however, the actual dimension of the system equation is seldom known. In a future work, one may optimize the rank of the system as well as the coefficients.

10 Summary

We proposed a new framework, DMD with memory, or, DMDm, a DMD-based numerical tool to analyze time-series data. Use of more generic linear operator instead of the finite difference operator enables us to take the memory effects into account in the time-evolution models. The memory effect is a widely seen phenomena in real world, e.g., viscoelastic matter and fluid dynamics. As one example of DMDm, we formulated fracDMD, whose name stands for *fractional* DMD, indicating the use of fractional-order derivative. The incorporation of fractional-order derivative in DMDm is equivalent to assume the power-law memory effects. We successfully show that using the fracDMD one can reconstruct the frequencies of fractional oscillators from noisy observations. The proposed method is expected to be useful to model unknown physical processes, such as thermal and mechanical processes in modern industrial machines.

Author Contributions

RA developed the proposed methods, formulated DMDm and fracDMD, and designed the numerical experiments. KS developed the numerical schemes (including the higher-order scheme for fracDMD) and built the numerical codes. Both of them performed numerical tests in Python and discussed on the results. TT, MK, and TM supervised the project and gave comments.

Conflict of Interest

We declare that we have no conflicts of interest.

Acknowledgments

We would like to thank our teammates for their comments on the theoretical and practical aspects of our methods. Mr. Takashi Yokota and Mr. Masaru Nishimura of Tokyo Electron Ltd. gave us valuable comments on the novelty of the proposed methods. We also thank their aids for patent application for the proposed methods. Our survey of existing research works are partly helped by the Access to Articles programme in Tokyo Electron Group.

References

[1] T. Askham and J. N. Kutz, *Variable projection methods for an optimized dynamic mode decomposition*, SIAM Journal on Applied Dynamical Systems, 17 (2018), pp. 380–416.

[2] S. L. Brunton, M. Budišić, E. Kaiser, and J. N. Kutz, *Modern koopman theory for dynamical systems*, arXiv preprint arXiv:2102.12086, (2021).

[3] S. Burov and E. Barkai, *Fractional langevin equation: Overdamped, underdamped, and critical behaviors*, Physical Review E, 78 (2008), p. 031112.

[4] H. J. Haubold, A. M. Mathai, and R. K. Saxena, *Mittag-leffler functions and their applications*, Journal of applied mathematics, 2011 (2011).

[5] D. Matignon, *Stability results for fractional differential equations with applications to control processing*, in Computational engineering in systems applications, vol. 2, Citeseer, 1996, pp. 963–968.

[6] I. Mezić, *Spectral properties of dynamical systems, model reduction and decompositions*, Nonlinear Dynamics, 41 (2005), pp. 309–325.

[7] I. Mezic, *Analysis of fluid flows via spectral properties of the koopman operator*, Annual Review of Fluid Mechanics, 45 (2013), pp. 357–378.

[8] I. Podlubny, *Fractional-order systems and pi/sup /spl lambda//d/sup /spl mu//-controllers*, IEEE Transactions on Automatic Control, 44 (1999), pp. 208–214, https://doi.org/10.1109/9.739144.

[9] J. L. Proctor, S. L. Brunton, and J. N. Kutz, *Dynamic mode decomposition with control*, SIAM Journal on Applied Dynamical Systems, 15 (2016), pp. 142–161.

[10] E. Rodrigues, B. Zadrozny, C. Watson, and D. Gold, *Decadal forecasts with resdmd: a residual dmd neural network*, arXiv preprint arXiv:2106.11111, (2021).

[11] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, *Spectral analysis of nonlinear flows*, Journal of fluid mechanics, 641 (2009), pp. 115–127.

[12] I. Sakata, T. Sakata, K. Mizoguchi, S. Tanaka, G. Oohata, I. Akai, Y. Igarashi, Y. Nagano, and M. Okada, *Complex energies of the coherent longitudinal optical phonon–plasmon coupled mode according to dynamic mode decomposition analysis*, Scientific Reports, 11 (2021), pp. 1–10.

[13] A. A. K. Samko, Stefan G. and O. I. Marichev, *Fractional integrals and derivatives: theory and applications*, Gordon and Breach Science Publishers, 1993.

[14] M. Sasaki, Y. Kawachi, R. Dendy, H. Arakawa, N. Kasuya, F. Kin, K. Yamasaki, and S. Inagaki, *Using dynamical mode decomposition to extract the limit cycle dynamics of modulated turbulence in a plasma simulation*, Plasma Physics and Controlled Fusion, 61 (2019), p. 112001.

[15] D. Sashidhar and J. N. Kutz, *Bagging, optimized dynamic mode decomposition for robust, stable forecasting with spatial and temporal uncertainty quantification*, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 380 (2022), p. 20210199, https://doi.org/10.1098/rsta.2021.0199.

[16] P. J. Schmid, *Dynamic mode decomposition of numerical and experimental data*, Journal of fluid mechanics, 656 (2010), pp. 5–28.

[17] P. J. Schmid, *Dynamic mode decomposition and its variants*, Annual Review of Fluid Mechanics, 54 (2022), pp. 225–254, https://doi.org/10.1146/annurev-fluid-030121-015835.

[18] A. A. Stanislavsky, *Fractional oscillator*, Phys. Rev. E, 70 (2004), p. 051103, https://doi.org/10.1103/PhysRevE.70.051103, https://link.aps.org/doi/10.1103/PhysRevE.70.051103.

[19] N. Sugimoto and T. Horioka, *Dispersion characteristics of sound waves in a tunnel with an array of helmholtz resonators*, Journal of the Acoustical Society of America, 97 (1995), pp. 1446–1459.

[20] A. Svenkeson, B. Glaz, S. Stanton, and B. J. West, *Spectral decomposition of nonlinear systems with memory*, Phys. Rev. E, 93 (2016), p. 022211, https://doi.org/10.1103/PhysRevE.93.022211.

[21] V. E. Tarasov, *General fractional dynamics*, Mathematics, 9 (2021), p. 1464.

[22] J. Tenreiro Machado and A. Azenha, *Fractional-order hybrid control of robot manipulators*, in SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No.98CH36218), vol. 1, 1998, pp. 788–793 vol.1, https://doi.org/10.1109/ICSMC.1998.725510.

[23] J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz, *On dynamic mode decomposition: Theory and applications*, Journal of Computational Dynamics, 1 (2014), pp. 391–421.