Springer

Acta Mechanica Sinica

CrossMark click for updates

• RESEARCH PAPER •

Acta Mech. Sin., Vol. 41, 524250 (2025)
https://doi.org/10.1007/s10409-025-24250-x

Model reconstruction of serial manipulators: a stepwise data-driven approach

Dingxu Guo[1], Jian Xu[1], Xiaoxu Zhang[2], Xiuting Sun[1], and Shu Zhang[1*]

[1] *School of Aerospace Engineering and Applied Mechanics, Tongji University, Shanghai 200092, China;*
[2] *Academy for Engineering & Technology, Fudan University, Shanghai 200433, China*

Received December 8, 2024; accepted January 2, 2025; published online April 11, 2025

Advancements in dynamic modeling methods of robotic manipulator are critical to the effective implementation of model-based control. Traditional approaches rely on rigorous first-principles-based dynamic modeling and precise parameter identification, while this paper explores an alternative through data-driven model reconstruction. To tackle the curse of dimensionality in the model reconstruction of a serial robotic manipulator with multi-degree-of-freedom, a relative activation indicator is proposed. Based on this indicator, the *k*-means clustering algorithm is utilized to classify the data under different working conditions. Subsequently, we leverage the fundamental prior knowledge to find the dynamical characteristics of each cluster and reconstruct the dynamic model in a stepwise manner using the method of sparse identification of nonlinear dynamics (SINDy). For the library generation of SINDy, the strategy of double-feature-set for serial manipulators with common joint types is proposed. Simulation results show that the stepwise model reconstruction approach not only reduces the size of the library of candidate functions but also decreases the impact of data noise on the reconstruction results. Finally, controllers based on the reconstructed models are deployed on the experimental platform and the experimental results demonstrate the improvement in trajectory tracking performance and the potential of the proposed method in engineering applications.

Serial manipulator, Dynamic modeling, Sparse identification, Relative activation indicator, Stepwise model reconstruction

Citation: D. Guo, J. Xu, X. Zhang, X. Sun, and S. Zhang, Model reconstruction of serial manipulators: a stepwise data-driven approach, Acta Mech. Sin. **41**, 524250 (2025), https://doi.org/10.1007/s10409-025-24250-x

1. Introduction

At present, serial robotic manipulators have been widely applied in various industrial fields, where the demands for high speed and precision have placed stringent standards on the control performance of such manipulators [1, 2]. Extensive research on robots has underscored the crucial significance of dynamics in guiding the design of model-based control laws, thereby improving control performance [3-5]. Reliable dynamic models can effectively leverage the advantages of model-based control whereas inaccurate models limit the application of model-based control methods to industrial serial manipulators.

In the traditional approach, obtaining a reliable dynamic model for a serial manipulator involves two main steps: dynamic modeling by means of first-principles and parameter identification based on the developed model. For multi-degree-of-freedom (multi-DoF) serial manipulators, the process of first-principles modeling requires expertise and inevitably involves ideal assumptions. Some factors present in serial manipulators, such as joint friction and nonlinearity with respect to control variables, greatly challenge the process of accurate modeling [6, 7]. Additionally, dynamic parameters of many robotic manipulators cannot be accurately provided and are difficult to measure directly, so a reliable dynamic model requires the identification of a bunch of parameters such as geometry, mass, rotational inertia, friction coefficient, etc [8-10]. On the basis of the dynamic model

*Corresponding author. E-mail address: zhangshu@tongji.edu.cn (Shu Zhang)
Executive Editor: Yanqing Wang

formulated by first-principles, many cases have successfully identified the dynamic parameters of robotic manipulators by designing optimized excitation trajectory and then estimating the parameters with least squares method [11-13]. However, the parameter identification process still relies on the formulated model [14], whose precision bounds the overall accuracy of theoretical prediction on dynamics [15]. In contrast, data-driven model reconstruction formulizes the dynamic behavior of the manipulator based on data rather than first-principles, avoiding the ideal assumptions as well as mathematical derivation and parameter identification in modeling process [16-18], and has received widespread attention in recent years.

Data-driven modeling approaches are generally classified into two categories in accordance with the form of the established models, i.e., machine learning methods leading to black-box models and symbolic regression methods leading to white-box analytical models [19-21]. The former have powerful fitting capabilities to handle complex nonlinear relationships in modeling robotic manipulators. However, the inherent black-box properties prevent the machine learning methods from providing a tangible physical perspective on the established models [22-24]. In contrast, the latter seeks to obtain an analytic model with physical interpretability, facilitating further theoretical analysis and model-based control evaluation [25]. For example, genetic programming (GP), as a symbolic optimization technique, enables the regression of symbolic models for nonlinear systems from data [26, 27]. Currently, GP-based symbolic regression methods have been successfully applied in various studies involving robotic manipulators, such as finding closed-form solutions for inverse kinematics [28], reconstructing the dynamic model [29], and designing the controllers [30]. However, the gradient-free learning process of GP poses often results in code bloat along with bizarrely complicated terms (e.g., $\sin(\sin(\sin x))$ [29]), hence prevents establishing a mathematically concise and physically meaningful model.

Sparse identification of nonlinear dynamics (SINDy) [31] is also an effective symbolic regression method of finding governing equations of dynamics of nonlinear systems. Sparse identification has been successfully applied in finding the variational law [32] and the Lagrangian from learned energy [33] hidden in physical systems, and reconstructing Jacobian matrix [34] and gravity term [35] of robotic manipulators. The core of SINDy lies in the integration of domain knowledge so as to construct a suitable library of candidate functions that serves to adequately describe the dynamic behavior of the system, ending up with a favorable model when the symbolic form of the target model is relatively concise. However, SINDy also suffers from excessively large library of candidate functions when dealing with high-dimensional nonlinear systems. Optimization methods such as SINDy-PI [36] have been proposed to tackle such issues. For instance, SINDy-PI has been successfully used for the library reduction in SINDy-based modeling of serial manipulators [37]. However, the number of DoF for the serial robotic manipulators addressed in Ref. [37] is stuck at two and consequently the main obstacle of applying SINDy is still not bypassed.

Data clustering, an unsupervised learning method facilitating grouping based on similarity or correlation in data, has been widely applied in data analysis, pattern recognition, and information processing [38-40]. Recently, researchers have recognized the potential of employing clustering algorithms in applications of symbolic regression. For example, Fernex et al. [41] introduced a cluster-based network method for data-driven modeling of complex nonlinear dynamics from time-resolved snapshot data. Additionally, Bramburger et al. [42] identified the slow timescale characteristics of a system with multiple timescales using clustering techniques and extracted the slow timescale dynamics via SINDy. Combined with unsupervised clustering techniques, the partition of certain library of candidate functions according to data features which arise from various working conditions may reduce the complexity of reconstructed models, and consequently provides a promising path to symbolic-regression-based modeling of multi-DoF serial manipulators.

This research aims to provide a stepwise data-driven approach for dynamic model reconstruction of multi-DoF serial manipulators. Compared with previous studies, the main contributions of this paper are as follows:

(1) A relative activation indicator (RAI) is introduced to classify the data features of multi-DoF serial manipulators, enabling the development of a stepwise SINDy method for dynamic model reconstruction.

(2) A double-feature-set library generation (DLG) method is proposed to enhance SINDy-based symbolic regression, reducing redundancy and generalizing for serial manipulators with common joint types.

(3) Simulations on a 4-DoF selective compliance assembly robot arm (SCARA) manipulator and a three-axis serial manipulator demonstrate the improvements of the stepwise SINDy in terms of library size and computational efficiency. Under noisy data conditions, the reconstructed model shows better sparsity and predictive accuracy.

(4) Experimental validation shows the effectiveness of the proposed method in accurately predicting torques and enhancing trajectory tracking, highlighting its potential for practical applications.

The remainder of this paper is organized as follows. In Sect. 2, the motivation of this paper is briefly introduced and the data clustering based on RAI is performed. Section 3 proposes the stepwise approach of model reconstruction,

i.e., stepwise SINDy, based on the clustering results, and two simulation platforms are employed to showcase advantages of the stepwise SINDy. Section 4 presents experimental results of a three-axis manipulator. The conclusion of this paper is presented in Sect. 5.

2. Data clustering

2.1 Motivation

The dynamic model of a serial manipulator, relating robot motion to joint driving torques, plays a crucial role in designing advanced control laws. For an n-DoF serial manipulator composed of rigid links, the dynamic equation can be derived through first principles, such as the Lagrangian method. Ignoring external disturbances and joint frictions, the dynamic model can be generally expressed in the following form:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\tau}, \tag{1}$$

where $\mathbf{q} \in \mathfrak{R}^n$, $\dot{\mathbf{q}} \in \mathfrak{R}^n$, $\ddot{\mathbf{q}} \in \mathfrak{R}^n$ are the vectors of generalized coordinates, generalized velocities, generalized accelerations, respectively; $\mathbf{M}$, $\mathbf{C}$, respectively represent the mass, the Coriolis and centrifugal matrices; $\mathbf{G}$ represents the gravity term; $\boldsymbol{\tau} \in \mathfrak{R}^n$ is the vector of generalized torques/forces. For the n-DoF serial manipulator, the process of first-principles modeling is tedious and skillful, and inevitably involves ideal assumptions. In contrast, SINDy-based model reconstruction formulizes the dynamic behavior of the manipulator using data, avoiding the ideal assumptions as well as mathematical derivation and parameter identification in modeling process.

The model of the n-DoF serial manipulator has the following property.

Property 1. The left side of Eq. (1) is linear with respect to system parameters [43], i.e.,

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\psi}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\,\theta, \tag{2}$$

where $\boldsymbol{\psi}$ represents the regression matrix and θ the vector of system parameters. The fundamental principle of SINDy is to sparsely represent a function as a linear combination of nonlinear terms. To this end, $\boldsymbol{\psi}$ is to be replaced by the libraries of candidate functions, namely

$$\boldsymbol{\psi}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\,\theta \rightarrow \mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\,\boldsymbol{\Xi}, \tag{3}$$

where $\mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$ is the library of candidate functions, $\boldsymbol{\Xi} = \{\boldsymbol{\Xi}_1,\ \boldsymbol{\Xi}_2,\ \ldots,\ \boldsymbol{\Xi}_n\}$ is the sparse matrix of coefficients. According to the characteristics of rotational joints and joint frictions of the serial manipulator, $\mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$ consists of constants, trigonometric functions, exponential functions, and polynomials. Subsequently, by recording the signals of robot motion and joint torques, we obtain

$$\begin{bmatrix} \tau_i(t_1) \\ \tau_i(t_2) \\ \vdots \end{bmatrix} = \begin{bmatrix} Y_1(t_1) & \cdots & Y_p(t_1) \\ Y_1(t_2) & \cdots & Y_p(t_2) \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \xi_{i1} \\ \xi_{i2} \\ \vdots \\ \xi_{ip} \end{bmatrix}, \tag{4}$$

where $\boldsymbol{\Xi}_i = \begin{bmatrix} \xi_{i1} & \xi_{i2} & \cdots & \xi_{ip} \end{bmatrix}^{\mathrm{T}}$ is the sparse vector of coefficients of ith joint, p the number of candidate functions, and $i = 1,\ 2,\ \ldots,\ n$. At last, $\boldsymbol{\Xi}$ is determined by solving Eq. (4) in different regression strategies.

As the DoF of the serial manipulator increases, more candidate functions with complicated form will appear in the library and the size of $\mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$ will grow exponentially, eventually leading to curse of dimensionality especially for the case where $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ all have significant measured values. Furthermore, signals of higher time-derivative measurement are more vulnerable regarding the signal-to-noise ratios (SNRs), implying that the regression may be severely disturbed by the noise if the signals of $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ are simultaneously taken into account for regression. In other words, separately collecting data according to different working conditions may facilitate efficient model reconstruction.

2.2 Data clustering based on RAI

Considering typical tasks of serial manipulators, stages with different data features will appear alternatively, indicating the possibility of quantifying such data features according to the activation of different signals. To this end, we first define the RAI as follows:

$$I(\mathbf{a}) = S(\mathbf{a}) + W\|\mathbf{a}\|_1 \operatorname{sign}\left(\sum_{i=1}^{n} a_i\right), \tag{5}$$

where $\mathbf{a} \in \mathfrak{R}^n$ can be substituted by the vector of generalized velocities $\dot{\mathbf{q}}$ or generalized accelerations $\ddot{\mathbf{q}}$, $\|\cdot\|_1$ is the sum of the magnitudes of the vector in space, and W is the weighting coefficient of the activation level. $S(\cdot)$ is the function of activation switch that takes the form

$$S(\mathbf{a}) = \tanh(h\|\mathbf{a}\|_1) = \frac{1 - \mathrm{e}^{-h\|\mathbf{a}\|_1}}{1 + \mathrm{e}^{-h\|\mathbf{a}\|_1}}, \tag{6}$$

where h is the coefficient that adjusts the sensitivity of the activation switch. After processing the generalized velocity and generalized acceleration signals with RAI, the collected data can be clustered using the k-means method. The procedure of the data clustering based on RAI is presented in Algorithm 1.

Algorithm 1 Procedure of data clustering based on RAI

Input: Acquire m sets of data, i.e., sample the data at m discrete moments
Data processing:
Assign values to W and h
Calculate $\mathbf{z}_i = \left[\, I(\dot{\mathbf{q}}(t_i))\ \ I(\ddot{\mathbf{q}}(t_i)) \,\right]^{\mathrm{T}}$ for $i = 1, 2, \ldots, m$
Data clustering: k-means method
Input: $\left\{ \mathbf{z}_1\ \ \mathbf{z}_2\ \ \cdots\ \ \mathbf{z}_m \right\}$, number of clusters k, and randomly initialize clustering centers $\left\{ \boldsymbol{\mu}_1\ \ \boldsymbol{\mu}_2\ \ \cdots\ \ \boldsymbol{\mu}_k \right\}$
Repeat
Let $\mathbb{C}_j = \emptyset,\ j = 1, 2, \ldots, k$
for $i = 1, 2, \ldots, m$ **do**
Calculate $d_{ij} = \left\| \mathbf{z}_i - \boldsymbol{\mu}_j \right\|_2,\ j = 1, 2, \ldots, k$
Let $\gamma = \arg\min_{j \in \{1, 2, \ldots, k\}} d_{ij}$, $\mathbb{C}_\gamma = \mathbb{C}_\gamma \cup \{\mathbf{z}_i\}$
end for
Let $r = 0$
for $j = 1, 2, \ldots, k$ **do**
Denote c_j as the number of elements of $\mathbb{C}_j$
Calculate the gravity position of $\mathbb{C}_j$:
$$\boldsymbol{\mu}_j' = \frac{\sum_{\mathbf{z} \in \mathbb{C}_j} \mathbf{z}}{c_j}$$
if $\boldsymbol{\mu}_j' \neq \boldsymbol{\mu}_j$ **then**
$\boldsymbol{\mu}_j = \boldsymbol{\mu}_j'$
else
$r = r + 1$
end if
end for
until $r = k$
Output: Classified clusters $\left\{ \mathbb{C}_1\ \ \mathbb{C}_2\ \ \cdots\ \ \mathbb{C}_k \right\}$

Figure 1 illustrates the process of data clustering based on RAI as sketched in Algorithm 1. Firstly, the n-DoF serial manipulator performs a sorting operation that involves typical states of the joints such as static, uniform velocity, and variable velocity. To simulate the noise in measurement, we add Gaussian White noise with SNRs of 60 and 50 dB to the signals of generalized velocities and generalized accelerations, respectively. Signals from each joint are collected during the operation of the robotic manipulator. Subsequently, the collected data is pre-processed by calculating RAI for the vector of generalized velocities $\dot{\mathbf{q}}$ and generalized accelerations $\ddot{\mathbf{q}}$

$$I_1 = I(\dot{\mathbf{q}}) = \frac{1 - \mathrm{e}^{-h\|\dot{\mathbf{q}}\|_1}}{1 + \mathrm{e}^{-h\|\dot{\mathbf{q}}\|_1}} + W_1 \|\dot{\mathbf{q}}\|_1 \operatorname{sign}\left(\sum_{i=1}^{n} \dot{q}_i \right), \tag{7}$$

and

$$I_2 = I(\ddot{\mathbf{q}}) = \frac{1 - \mathrm{e}^{-h\|\ddot{\mathbf{q}}\|_1}}{1 + \mathrm{e}^{-h\|\ddot{\mathbf{q}}\|_1}} + W_2 \|\ddot{\mathbf{q}}\|_1 \operatorname{sign}\left(\sum_{i=1}^{n} \ddot{q}_i \right), \tag{8}$$

where the parameters are chosen as $h = 60$, $W_1 = 0.6$, and $W_2 = 0.32$. Finally, the k-means method is employed to cluster the processed data as detailed in Algorithm 1. It's worth noting that the number of clusters k is selected by the elbow method. Here the within-cluster sum-of-squares (WCSS) is introduced as an evaluation index of clustering, which is defined as [44]

$$WCSS = \sum_{i=1}^{k} \sum\nolimits_{\mathbf{z} \in \mathbb{C}_i} \left\| \mathbf{z} - \boldsymbol{\mu}_i \right\|_2, \tag{9}$$

where $\|\cdot\|_2$ is the distance of the vector coordinate from the origin of the vector space. As shown in Fig. 1, the elbow point of the number of clusters arises when $k = 3$. Additionally, Algorithm 1 is run 20 times with randomly initialized clustering centers, and the maximum number of iterations is 12 while the average number of iterations is 5.75, indicating the effectiveness of the data clustering.

Remark. The reason that RAI defined by Eq. (5) is termed as "relative" is as follows. The first term on the right-hand side of Eq. (5), i.e., $S(\mathbf{a})$, appears as a filter that determines which signal is considered as activated, and the second term quantifies the activation level based on which the post-clustering data is uniformly distributed around $S(\mathbf{a})$. Therefore, Eq. (5) indicates the relative magnitude of the activation. However, it should be noted that the first term in RAI is still dominant if a proper value is assigned to W.

2.3 Physical interpretation of clustering patterns

In this subsection, we aim to elucidate the physical interpretation of the clustering results observed in the last subsection. Backtracking from three clusters to collected data, they respectively represent the static state, constant velocity motion, and variable velocity motion as shown in Fig. 2. Data in the first cluster leads to the least significant RAI and depends almost only on generalized coordinates, implying that the library of candidate functions that reconstructs the model for the static case has the simplest form and the smallest size. Meanwhile, data in the second and third clusters activate respectively the functions involving generalized velocities and generalized accelerations, indicating that the complexity of candidate functions will be growing in the corresponding cases.

Revisiting the dynamic equation of the serial manipulator and ignoring the joint frictions, the motion signals in the three cases, namely, the static state, motion with constant velocity, and motion with variable velocity, will sequentially activate the gravity terms, the Coriolis and centrifugal terms, and the inertia terms, respectively. Consequently, reconstructing the dynamic model of the serial manipulator in a stepwise manner, based on the recognition of the aforementioned cases, can effectively reduce the complexity of symbolic regression for (i) the regression results of the current step can be inherited by the subsequent step, and (ii) the complexity of matrix operations, essentially abundant in the SINDy regression, can be significantly reduced. Furthermore, since the signals of

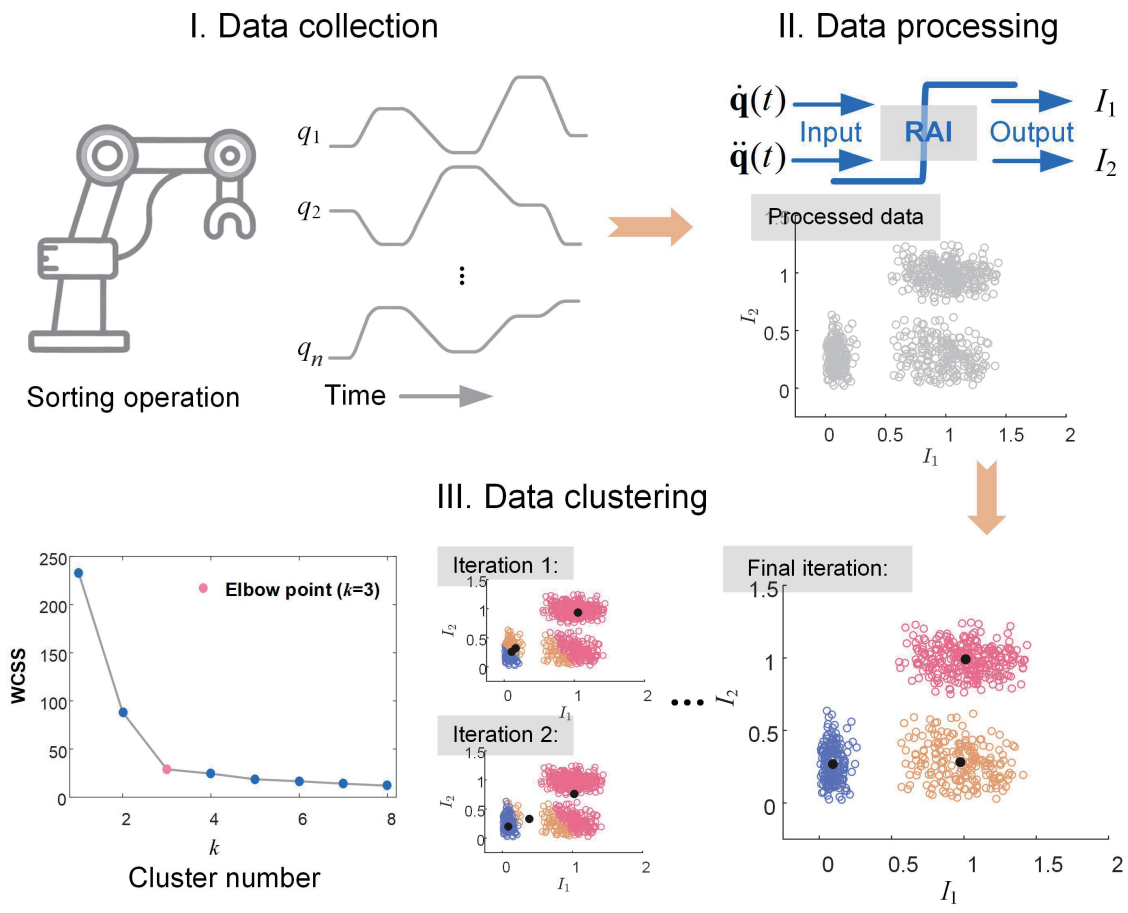

Figure 1 Flow diagram of data clustering based on activation indicator where I is the data collection of the multi-DoF serial manipulator, II the processing of the collected motion signals using RAI, and III the clustering of the processed data using the k-means method.

generalized velocity and generalized acceleration are separately used in different steps, this approach can effectively mitigate the impact of noise on the reconstruction results.

3. Stepwise model reconstruction

Previous analyses and clustering results have shown the feasibility of stepwise reconstruction of the dynamic model of a serial manipulator. In this section, we will illustrate the stepwise model reconstruction approach in terms of library generation, data collection, and regression strategy.

3.1 Double-feature-set library generation

Combined with the domain knowledge, the library of candidate functions i.e., $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$, is assumed to be constructed by the elementary functions including generalized velocities, generalized accelerations, generalized coordinates, and trigonometric functions of generalized coordinates as building blocks. Due to the kinematic constraints inherent in multi-DoF serial manipulators, expressions concerning generalized coordinates in the dynamic model typically exhibit greater complexity when contrasted with expressions in terms of generalized velocity and generalized acceleration. To fully describe the constituents of library and reduce non-essential candidate functions, DLG is proposed. For the library generation of SINDy, the first feature set is given as

$$\mathcal{F}_1 = \left\{ \prod_{i=1}^{N_1} c_{1i} \right\}. \tag{10}$$

In Eq. (10), c_{1i} is the elementary function, the basic components for feature set $\mathcal{F}_1$, and

$$c_{1i} \in \{1\} \cup \{E_1\} \cup \{E_2\} \cdots \cup \{E_n\}, \tag{11}$$

where $E_i = \{\sin q_i, \cos q_i\}$ if the ith joint is a revolute joint and $E_i = \{q_i\}$ if the ith joint is a prismatic joint. N_1 represents truncated order of the elementary functions, which determine the complexity of features in $\mathcal{F}_1$. Since the kinematic relations are simpler in the latter, prismatic joints are more friendly to library generation. {1} in Eq. (11) is to ensure that

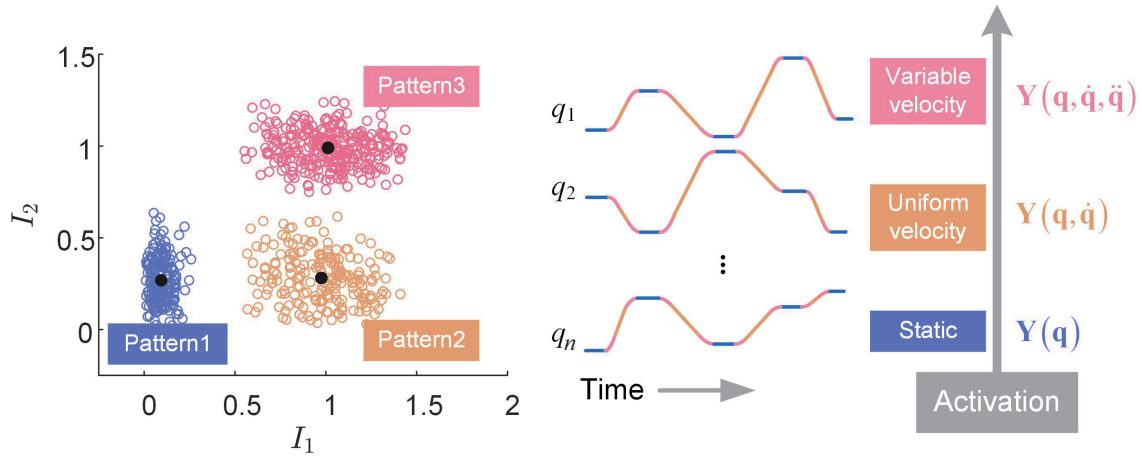

Figure 2 Physical interpretation of clustering results.

as N_1 increases, the feature set $\mathcal{F}_1$ still contains some simple terms. For instance, if $N_1 = 3$ in Eq. (10), the complex features in $\mathcal{F}_1$ may be $\left\{\sin q_1 \cos^2 q_2\right\}$ and the simplest feature is $\{1\}$. We define

$$\alpha_1 = |\{1\} \cup \{E_1\} \cup \{E_2\} \cdots \cup \{E_n\}|, \tag{12}$$

where $|\cdot|$, called cardinality of the set, denotes the number of elements of the set. The number of features (i.e., elements) in $\mathcal{F}_1$ can be calculated as

$$|\mathcal{F}_1| = 1 + \sum_{i=1}^{N_1} C_{\alpha_1+i-1}^{i}. \tag{13}$$

Then, the second feature is given as

$$\mathcal{F}_2 = \left\{\prod_{i=1}^{N_2} c_{2i}\right\}, \tag{14}$$

where

$$c_{2i} \in \{1\} \cup \{\dot{q}_1, \dot{q}_2, \ldots, \dot{q}_n\} \cup \{\ddot{q}_1, \ddot{q}_2, \ldots, \ddot{q}_n\}. \tag{15}$$

We define

$$\alpha_2 = |\{1\} \cup \{\dot{q}_1, \dot{q}_2, \ldots, \dot{q}_n\} \cup \{\ddot{q}_1, \ddot{q}_2, \ldots, \ddot{q}_n\}|. \tag{16}$$

The number of features in $\mathcal{F}_2$ can be calculated as

$$|\mathcal{F}_2| = 1 + \sum_{i=1}^{N_2} C_{\alpha_2+i-1}^{i}. \tag{17}$$

Finally, the complete feature set then has the following form:

$$\mathcal{F} = \mathcal{F}_1\mathcal{F}_2 := \left\{\left(\prod_{i=1}^{N_1} c_{1i}\right)\left(\prod_{i=1}^{N_2} c_{2i}\right)\right\}. \tag{18}$$

Each term on the right-hand side of Eq. (2) can be represented by a linear combination of the features (i.e., elements) in $\mathcal{F}$, therefore all features in $\mathcal{F}$ make up the library of candidate functions. The procedure of DLG is presented in Algorithm 2. DLG excludes non-essential candidate functions containing higher-order generalized velocities and higher-order generalized accelerations, which is beneficial for both SINDy and the stepwise SINDy.

Algorithm 2 Procedure of DLG

Input: Truncated orders N_1 and N_2
Generate feature sets $\mathcal{F}_1$ and $\mathcal{F}_2$
Let $a = |\mathcal{F}_1|$, $b = |\mathcal{F}_2|$, $\mathcal{F} = \emptyset$
for $j = 1, 2, \ldots, a$ **do**
  for $k = 1, 2, \ldots, b$ **do**
    $\mathcal{F} = \mathcal{F} \cup \{\mathcal{F}_1(j) \cdot \mathcal{F}_2(k)\}$
  end for
end for
$\mathbf{Y} = \begin{bmatrix} | & | & | & | \\ \mathcal{F}(1) & \mathcal{F}(2) & \cdots & \mathcal{F}(a \cdot b) \\ | & | & | & | \end{bmatrix}$
Output: Library $\mathbf{Y}$

3.2 DLG for stepwise SINDy

Inspired by the clustering results together with the terms appearing on the left-hand side of Eq. (2), we conclude that the model of the n-DoF serial manipulator has the following property.

Property 2. The right-hand side of Eq. (2) can be rewritten as

$$\boldsymbol{\psi}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\,\boldsymbol{\theta} = \underbrace{\boldsymbol{\psi}_3(\mathbf{q},\ddot{\mathbf{q}})\,\boldsymbol{\theta}_3}_{\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}} + \underbrace{\boldsymbol{\psi}_2(\mathbf{q},\dot{\mathbf{q}})\,\boldsymbol{\theta}_2}_{\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}} + \underbrace{\boldsymbol{\psi}_1(\mathbf{q})\,\boldsymbol{\theta}_1}_{\mathbf{G}(\mathbf{q})}. \tag{19}$$

With this, the dynamic model of the n-DoF serial manipulator can be reconstructed in three steps. Herein, we denote

the symbols in the ith step by $\{\cdot\}^{S_i}$ with $i = 1, 2, 3$. In SINDy-based model reconstruction, $\boldsymbol{\psi}_i$ is replaced by the libraries of candidate functions, namely

$$\boldsymbol{\psi}_i \theta_i \to \mathbf{Y}^{S_i} \boldsymbol{\Xi}^{S_i}, \quad i = 1,\ 2,\ 3. \tag{20}$$

When performing DLG in stepwise SINDy, elementary functions of the first feature set is the same as Eq. (11) and we have

$$\mathcal{F}_1^{S_i} = \left\{ \prod_{i=1}^{N_1^{S_i}} c_{1i} \right\}, \quad i = 1,\ 2,\ 3. \tag{21}$$

Meanwhile, elementary functions of the second feature set require a specific design for each step.

(1) Step I

In the first cluster, data with the least significant RAI depends almost only on generalized coordinates, implying that the library of candidate functions that reconstructs the model for the static case has the simplest form and the smallest size. Therefore, we choose the data collected from the static case for the first step of the model reconstruction. Then we have

$$\boldsymbol{\psi}\theta = \underbrace{\boldsymbol{\psi}_1(\mathbf{q})\,\theta_1}_{\mathbf{G}(\mathbf{q})}. \tag{22}$$

In this step, we use the following replacement:

$$\underbrace{\boldsymbol{\psi}_1(\mathbf{q})\,\theta_1}_{\mathbf{G}(\mathbf{q})} \to \mathbf{Y}^{S_1} \boldsymbol{\Xi}^{S_1}. \tag{23}$$

Therefore, the elementary function of the second feature set in Step I is selected as $c_{2i}^{S_1} = \{1\}$, and the second feature set is determined as

$$\mathcal{F}_2^{S_1} = \left\{ \prod_{i=1}^{N_2^{S_1}} c_{2i}^{S_1} \right\} = \{1\}. \tag{24}$$

(2) Step II

In the second cluster, the data activates functions involving generalized velocities. The data of motion with uniform velocities is collected for the second step of the model reconstruction. Then it is clear that

$$\boldsymbol{\psi}\theta = \underbrace{\boldsymbol{\psi}_2(\mathbf{q}, \dot{\mathbf{q}})\,\theta_2}_{\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}} + \underbrace{\boldsymbol{\psi}_1(\mathbf{q})\,\theta_1}_{\mathbf{G}(\mathbf{q})}, \tag{25}$$

where $\boldsymbol{\psi}_1(\mathbf{q})\,\theta_1$ is inherited from the regression results of the first step. In this step, we use the following replacement:

$$\underbrace{\boldsymbol{\psi}_2(\mathbf{q}, \dot{\mathbf{q}})\,\theta_2}_{\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}} \to \mathbf{Y}^{S_2} \boldsymbol{\Xi}^{S_2}, \tag{26}$$

with a particular form of the second feature set given as

$$\mathcal{F}_2^{S_2} = \left\{ \prod_{i=1}^{N_2^{S_2}} c_{2i}^{S_2} \right\}, \tag{27}$$

where $c_{2i}^{S_2} \in \{1\} \cup \{\dot{q}_1,\ \dot{q}_2,\ \ldots,\ \dot{q}_n\}$.

(3) Step III

In the third cluster, the data further activates the functions involving generalized accelerations. The data of motion with variable velocities is collected for the final step of the model reconstruction. It is clear that

$$\boldsymbol{\psi}\theta = \underbrace{\boldsymbol{\psi}_3(\mathbf{q}, \ddot{\mathbf{q}})\,\theta_3}_{\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}} + \underbrace{\boldsymbol{\psi}_2(\mathbf{q}, \dot{\mathbf{q}})\,\theta_2}_{\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}} + \underbrace{\boldsymbol{\psi}_1(\mathbf{q})\,\theta_1}_{\mathbf{G}(\mathbf{q})}, \tag{28}$$

where $\boldsymbol{\psi}_1(\mathbf{q})\,\theta_1$ and $\boldsymbol{\psi}_2(\mathbf{q}, \dot{\mathbf{q}})\,\theta_2$ are inherited from the regression results of the first and second steps, respectively. In this step, we use the following replacement:

$$\underbrace{\boldsymbol{\psi}_3(\mathbf{q}, \ddot{\mathbf{q}})\,\theta_3}_{\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}} \to \mathbf{Y}^{S_3} \boldsymbol{\Xi}^{S_3}, \tag{29}$$

with a particular form of the second feature set given as

$$\mathcal{F}_2^{S_3} = \left\{ \prod_{i=1}^{N_2^{S_3}} c_{2i}^{S_3} \right\}, \tag{30}$$

where $c_{2i}^{S_3} \in \{1\} \cup \{\ddot{q}_1,\ \ddot{q}_2,\ \ldots,\ \ddot{q}_n\}$.

3.3 Model reconstruction

In this subsection, the complete procedure of model reconstruction for the n-DoF serial manipulator is presented, and the flow diagram is shown in Fig. 3. At first, signals conforming to the three different clustering patterns are collected separately after determining the range and number of data collections, and the strategies for the data collections are proposed. In the data collection of the third clustering pattern, the form of Fourier-series-based trajectories is given as

$$q_i(t) = k_1 \cos(\omega_1 t) + k_2 \sin(\omega_2 t) + k_3 \cos(\omega_3 t), \tag{31}$$

where $i = 1,\ 2,\ \ldots,\ n$, k and ω are randomly generated. Next, the optimal truncated orders of feature sets need to be determined. The guiding principle for selecting the optimal truncation order is to minimize it while ensuring an adequate description of the dynamic behavior of the system, and we will give examples in the next section. Then, libraries of SINDy and stepwise SINDy are generated by using DLG. Finally, the reconstructed models of SINDy and

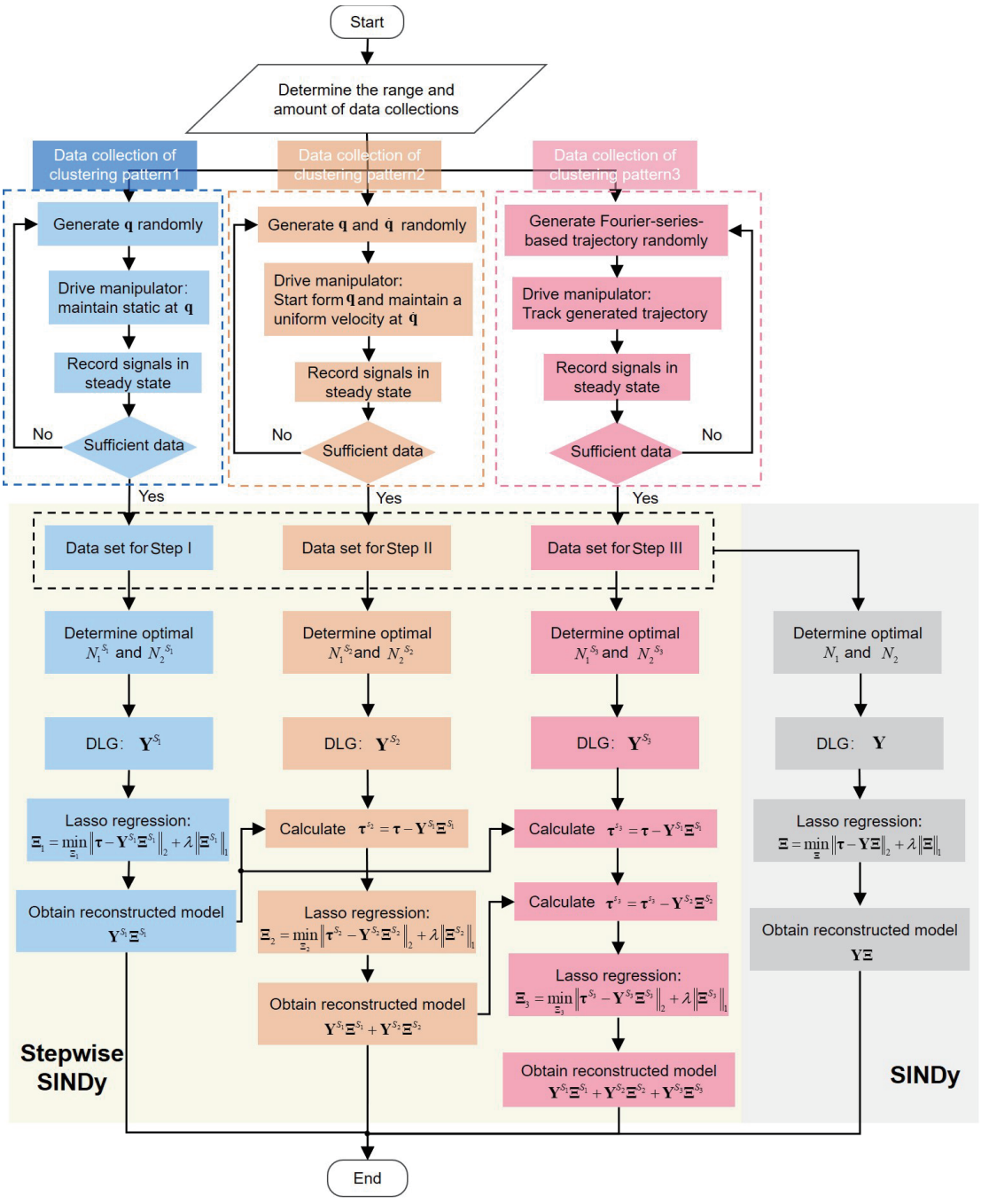

Figure 3 Flow diagram of the dynamic model reconstruction for the n-DoF serial manipulator.

stepwise SINDy are obtained by solving the regression problem. Herein, the Lasso regression with L1 norm penalties/constraints [45] is employed for finding sparse solutions to regression problems. The problem is formulated as

$$\Xi = \min_{\Xi} \|\boldsymbol{\tau} - \mathbf{Y}\Xi\|_2 + \lambda\|\Xi\|_1, \tag{32}$$

where λ is the sparsity promoting parameter. For stepwise model reconstruction, the reconstructed model obtained in the current step needs to be passed along to the next step, and the reconstructed model in the ith step is given by

$$\sum_{j=1}^{i} \mathbf{Y}^{S_j}\Xi^{S_j},\ i = 1,\ 2,\ 3. \tag{33}$$

4. Method validation

In this section, two simulation platforms, including a 4-DoF SCARA manipulator with a prismatic joint and a three-

axis serial manipulator, are employed to demonstrate the advantages of the proposed stepwise model reconstruction approach. In addition, the proposed approach is further validated on a three-axis experimental platform. The details are as follows.

4.1 Simulation of SCARA manipulator

SCARA is a prevalent industrial robot system that is widely recognized for its high rigidity and positioning accuracy in the horizontal plane [46]. The SCARA robot features a kinematic structure where the first two revolute joints have parallel axes, enabling planar motion of the end effector in the horizontal plane. Additionally, the vertical position and orientation of the end effector are independently controlled by the third prismatic joint and the fourth revolute joint, resulting in minimal kinematic coupling among the four DoFs and a simplified dynamic model. In this subsection, a typical 4-DoF SCARA manipulator with three revolute joints and one prismatic joint is presented as the first example to test the performance of the proposed stepwise model reconstruction approach. The configuration and parameters of the SCARA manipulator are shown in Fig. 4 where the simulations of inverse dynamics are performed in MATLAB-Robotic Toolbox.

The dynamic model of the SCARA manipulator is reconstructed following the procedure shown in Fig. 3. At first, the motion range of data collections is prescribed as shown in Table 1, and the numbers of data collected for the stepwise SINDy across three steps are 2000, 4000, and 4000, respectively. Then, for the SCARA manipulator with a simple dynamic model, the optimal truncated orders of the feature sets are small and can be selected in accordance with the guiding principle. The comparison between SINDy and stepwise SINDy of the SCARA manipulator is shown in Table 2. Under the optimal truncation orders, the total number of candidate functions for stepwise regression is 168, and for SINDy is 360, representing a decrease of 53.33%. The reduction in the size of the library of candidate functions is accompanied by an improvement in computational efficiency. Performing a single calculation of least squares regression, the total time consumption for the stepwise SINDy is 0.1165 s, whereas for SINDy it is 1.48 s, representing a decrease of 92.13%.

In addition, we study the performance of the stepwise SINDy in the presence of data noise. To simulate the noise in measurement, Gaussian White noise with SNRs of 60, 40, and 20 dB are added to generalized coordinates, generalized velocities, and generalized accelerations, respectively. The sparse vectors of coefficients, i.e., $\boldsymbol{\Xi}_i$, $i = 1, 2, 3, 4$, are solved from Eq. (32) where λ is chosen to be 0.02. Components in $\boldsymbol{\Xi}_i$ with an absolute value exceeding 0.001 are considered as the active items. As shown in Fig. 5, the number of active items in the reconstructed model of the stepwise SINDy is significantly reduced compared to that of SINDy, by 87.1% for $\boldsymbol{\Xi}_1$, 64.3% for $\boldsymbol{\Xi}_2$, 88.6% for $\boldsymbol{\Xi}_3$, and 88.2% for $\boldsymbol{\Xi}_4$. The fewer number of active items indicates that the reconstructed model is more concise, which generally makes the model more predictive. To compare the predictive ability of the reconstructed models of the two methods, a trajectory outside the range of data collection is used for torque/force

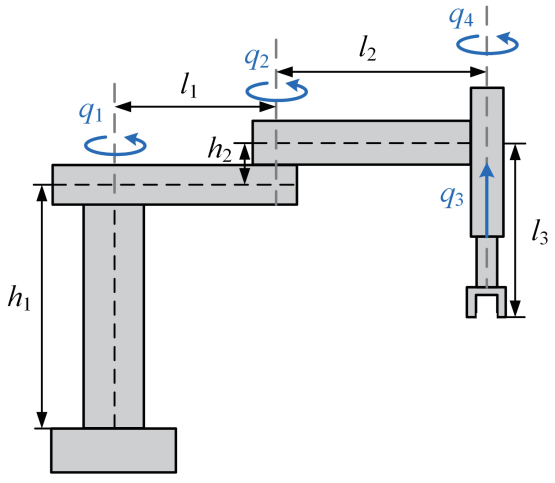

Figure 4 Geometric model of the SCARA manipulator. The geometrical parameters of the mechanism are chosen as $h_1 = 0.12$ m, $h_2 = 0.03$ m, $l_1 = 0.18$ m, $l_2 = 0.14$ m, and $l_3 = 0.1$ m. From base to end, the masses of the three homogenized links are 1.98, 1.95, and 0.2 kg, respectively, and the rotational inertia of the third link is 0.07 kg · m^2.

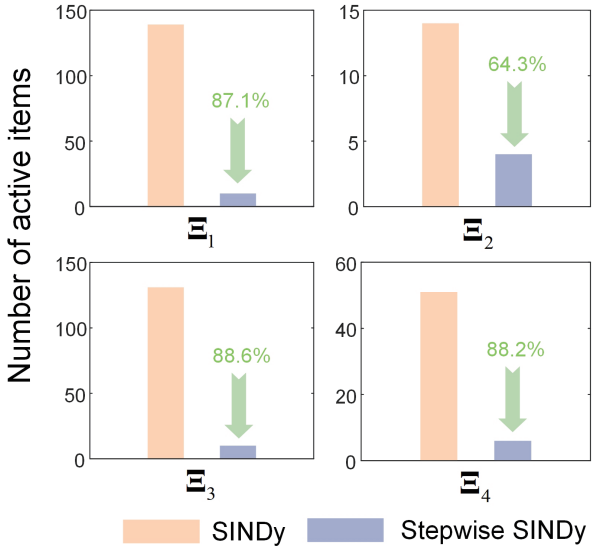

Figure 5 Comparison of active items of reconstructed models obtained by SINDy and stepwise SINDy, where the numbers of active terms in $\boldsymbol{\Xi}$ for SINDy and the sum of the numbers of active terms in $\boldsymbol{\Xi}^{S_1}$, $\boldsymbol{\Xi}^{S_2}$, and $\boldsymbol{\Xi}^{S_3}$ for the stepwise SINDy.

Table 1 Motion range of the SCARA manipulator in data collections

Joint	Generalized coordinate	Generalized velocity	Generalized acceleration
1	$[-\pi/2\ \ \pi/2]$ (rad)	$[-\pi/4\ \ \pi/4]$ (rad/s)	$[-\pi/2\ \ \pi/2]$ (rad/s^2)
2	$[-\pi/2\ \ \pi/2]$ (rad)	$[-\pi/4\ \ \pi/4]$ (rad/s)	$[-\pi/2\ \ \pi/2]$ (rad/s^2)
3	$[0\ \ 0.2]$ (m)	$[-0.1\ \ 0.1]$ (m/s)	$[-0.2\ \ 0.2]$ (m/s^2)
4	$[-\pi/2\ \ \pi/2]$ (rad)	$[-\pi/4\ \ \pi/4]$ (rad/s)	$[-\pi/2\ \ \pi/2]$ (rad/s^2)

Table 2 Comparison between SINDy and the stepwise SINDy of the SCARA manipulator

Method	Optimal N_1	Optimal N_2	Number of candidate functions	Number of data	Computational time-consuming (s)
SINDy	1	2	360	10000	1.48
Step I	1	0	8	2000	0.0015
Step II	1	2	120	4000	0.09
Step III	1	1	40	4000	0.025

predictions. As shown in Fig. 6, the reconstructed model using the stepwise SINDy provides a better predictive ability than the reconstructed model using SINDy.

4.2 Simulation of three-axis serial manipulator

To conveniently compare with the experimental results obtained from a serial manipulator with three revolute joints, a three-axis serial manipulator is introduced to validate the effectiveness of the stepwise model reconstruction. The diagram and parameters of the three-axis serial manipulator are shown in Fig. 7 where the simulations of inverse dynamics are performed in MATLAB-Robotic Toolbox.

Similarly, the dynamic model of the three-axis manipulator is reconstructed following the procedure shown in Fig. 3. The motion range of data collections is presented in Table 3 and the numbers of data points collected for the stepwise SINDy across three steps are 1000, 10000, and 10000, respectively. Despite having only three degrees of freedom, the three-axis serial manipulator involves multidimensional motions, thus leading to a dynamic model more complicated than that of the SCARA manipulator. To find the optimal truncated orders, we generate libraries of candidate functions for SINDy and the stepwise SINDy at different truncated orders, and then calculations of least squares regression for the collected data sets are performed. The regression errors at different orders are shown in Fig. 8. The figures show that the errors of least squares regression decrease rapidly when the orders reach a certain value, and no longer decrease remarkably when the orders keep increasing, implying the existence of optimal truncated orders. The comparison between SINDy and stepwise SINDy of the three-axis manipulator is

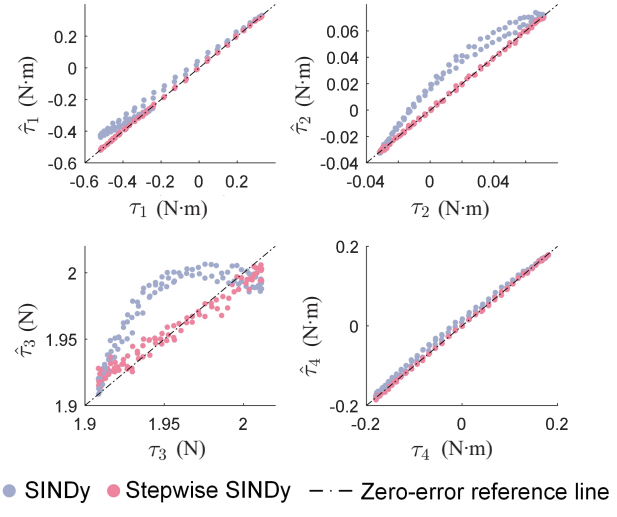

Figure 6 Comparison of the predictive ability of reconstructed models obtained by SINDy and stepwise SINDy, where torque/force predictions for a trajectory outside the range of data collections where the trajectory is given by $q_i(t) = \frac{\pi}{2} + \frac{\pi}{12}\sin(\pi t)$, $i = 1, 2, 4$, and $q_3(t) = 0.2 + 0.05\sin(\pi t)$.

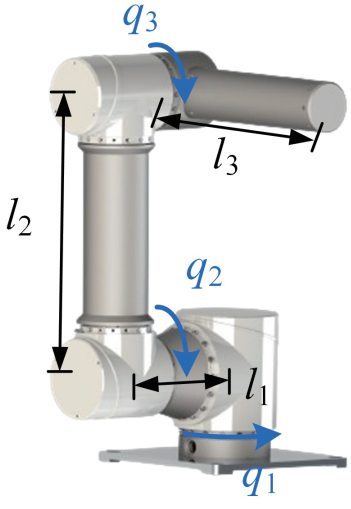

Figure 7 CAD model of the three-axis serial manipulator. From base to end, the lengths and masses of the three homogenized links are 0.2, 0.4, 0.35 m and 0.5, 0.83, 0.72 kg, respectively, and the masses of the three joint motors are 0.97, 0.43, 0.43 kg.

Table 3 Motion range of the three-axis manipulator in data collections

Joint	Generalized coordinate (rad)	Generalized velocity (rad/s)	Generalized acceleration (rad/s^2)
1	$[-\pi/2\ \ \pi/2]$	$[-\pi/2\ \ \pi/2]$	$[-\pi\ \ \pi]$
2	$[-\pi/3\ \ \pi/3]$	$[-\pi/2\ \ \pi/2]$	$[-\pi\ \ \pi]$
3	$[-\pi\ \ \pi]$	$[-\pi/2\ \ \pi/2]$	$[-\pi\ \ \pi]$

shown in Fig. 9. Under the optimal truncation orders, the total number of candidate functions for stepwise regression is 2968 and SINDy is 5880, representing a decrease of 49.52%. Performing a single calculation of least squares regression, the total time consumption for the stepwise SINDy is 32.3 s, whereas for SINDy it is 499.9 s, representing a decrease of 93.54%.

Furthermore, we discuss the performance of the stepwise SINDy under data noise through the following three cases. In Case I, all collected data is noise-free. In Case II, we add Gaussian White noise with SNRs of 70, 50, and 30 dB to the signals of generalized coordinates, generalized velocities, and generalized accelerations, respectively. In Case III, the collected data is added with stronger Gaussian White noise with SNRs of 60, 40, and 20 dB to the signals of generalized coordinates, generalized velocities, and generalized accelerations, respectively. The sparse vectors of coefficients, i.e., $\boldsymbol{\Xi}_i$, $i = 1, 2, 3$, are solved from Eq. (32) where λ is chosen to be 1. Components in $\boldsymbol{\Xi}_i$ with an absolute value exceeding 0.01 are considered as the active items. As shown in Fig. 10, the number of active items in the regression results of the stepwise SINDy is close to that of the traditional SINDy in Case I, indicating an equal level of sparsity of applying the two methods in the absence of noise. In Case II, the number of active items in the regression results of the stepwise SINDy is significantly reduced compared to that of the traditional SINDy, by 75.62% for $\boldsymbol{\Xi}_1$, 83.81% for $\boldsymbol{\Xi}_2$, and 50.00% for $\boldsymbol{\Xi}_3$. With the enhancement of the noise as in Case III, the sparsity of regression results of stepwise SINDy turns out to be more pronounced as reductions of 83.34% for $\boldsymbol{\Xi}_1$, 85.57% for $\boldsymbol{\Xi}_2$, and 85.78% for $\boldsymbol{\Xi}_3$ are observed. A reduced number of active items in the reconstructed model signifies a higher level of sparsity, thereby typically enhancing its predictive capability. To compare the predictive ability of the reconstructed models of the two methods, a trajectory outside the range of data collection is used for torque predictions. As

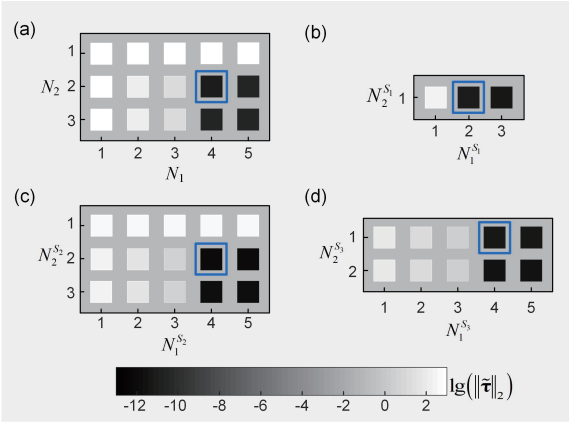

Figure 8 Optimal orders of the elementary functions in the feature sets of SINDy and stepwise SINDy: regression errors at different orders for (a) SINDy, (b) Step I of stepwise SINDy, (c) Step II of stepwise SINDy, and (d) Step III of stepwise SINDy, where the optimal orders are marked in blue boxes.

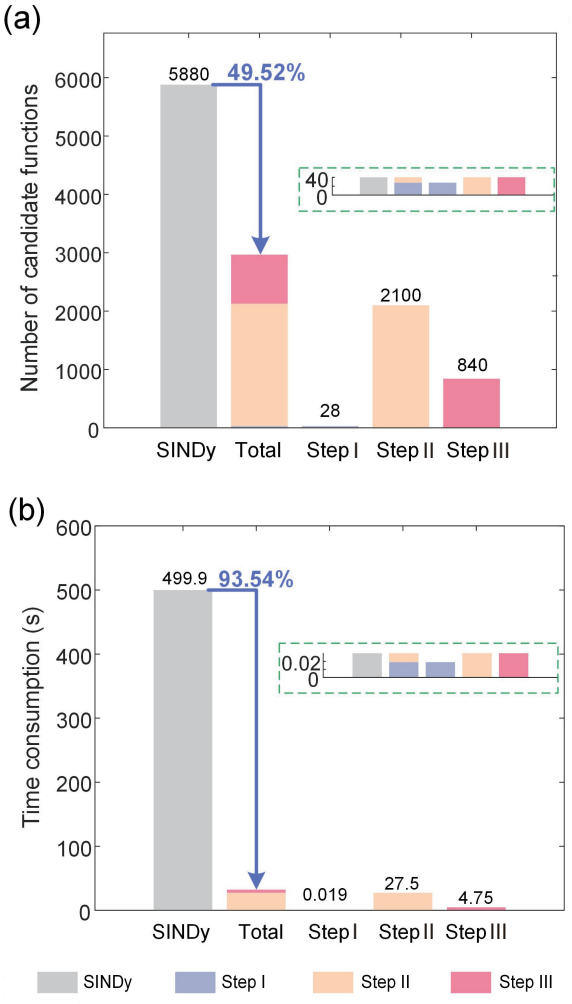

Figure 9 Comparisons between the two methods with respective optimal truncated orders in (a) the number of candidate functions and (b) the computational efficiency of regression where the regions magnified in the figure are enclosed within the green boxes.

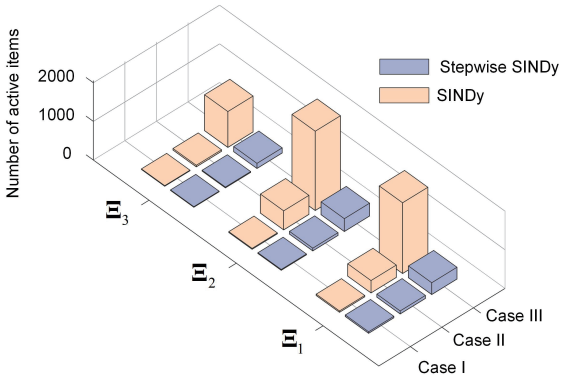

Figure 10 Numbers of active items in regression results of the two methods for cases with different noise intensities, where the numbers of active terms in Ξ for SINDy and the sum of the numbers of active terms in Ξ^{S_1}, Ξ^{S_2}, and Ξ^{S_3} for the stepwise SINDy.

shown in Fig. 11, the reconstructed model using the stepwise SINDy still provides promising prediction in Cases II and III, while the reconstructed model using the traditional SINDy fails to accurately predict the torques in Case III with the strong noise.

4.3 Experiment of three-axis serial manipulator

As shown in Fig. 12, an experimental platform of three-axis serial manipulator is built to further validate the effectiveness of the proposed method under realistic working conditions. The manipulator is equipped with three Elephant$^{@}$ joint modules, each of which is integrally composed of a servopack, encoder, and reducer. The produce numbers of joint modules are MS32 for joint 1 and MS20 for joints 2 and 3. The SpeedGoat$^{@}$ Unit is implemented as the programmable controller. In the experiment, we collect data for model reconstruction and verification in the speed mode of the joint modules. Based on the reconstructed model, we design the control law which is experimentally deployed in the torque mode to guarantee the performance of trajectory tracking. The sampling frequency of joint angles and joint velocities is 1000 Hz.

In the experiment, one has to consider the friction in the joint module in which the reducer may cause complicated frictional behaviors. In addition to the aforementioned procedure of data-driven model reconstruction (as shown in Fig. 3), the friction is modeled by the typical Stribeck model [47], given by

$$f(v) = \left[f_c + (f_s - f_c)\,\mathrm{e}^{-(v/v_s)^2}\right]\mathrm{sign}(v) + f_v v. \tag{34}$$

Note that the sign function and exponential function in Eq. (34) are difficult to handle in constructing the library of candidate functions. For this reason, we include in the model reconstruction the following treatments: (i) for the sign function, we categorize the collected data into the forward stage and backward stage according to the direction of the joint motions and then perform the stepwise SINDy for either

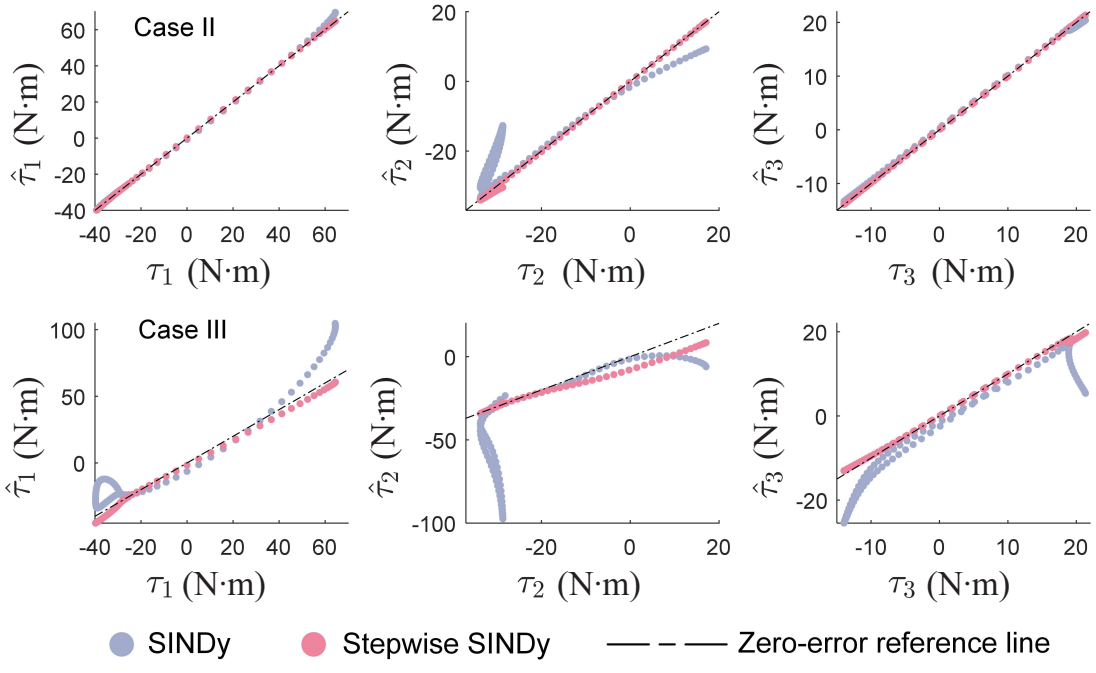

Figure 11 Torque predictions for a trajectory outside the range of data collections where the trajectory is given by $q_i(t) = \frac{\pi}{2} + \frac{\pi}{3}\sin(\pi t)$, $i = 1, 3$, and $q_2(t) = \frac{\pi}{6}\sin(\pi t) - \frac{\pi}{2}$. The figures in the top row are the torque predictions for Case II, and those in the bottom row are the torque predictions for Case III.

stage of the data, and (ii) for the exponential function, we add exponential functions of joint velocities with different coefficients, i.e., $e^{-(\dot{q}_i/v_s)^2}$ where $i = 1, 2, 3$ and $v_s = 0.01, 0.02, \ldots, 1$, to the library of candidate functions. It is worth mentioning that even in the static state the friction may also incorporate with the gravity, resulting in the difficulty of experimentally acquiring effective data for Step I in the stepwise SINDy. Alternatively, in this step the joint modules are actuated at a very low constant speed, i.e., 0.1 °/s, to overcome the coulomb dry friction. As a complement to Fig. 3, Fig. 13 illustrates the procedures of data collection and model reconstruction for the first two steps of stepwise SINDy in the experiment.

The reconstructed models are employed to predict torques

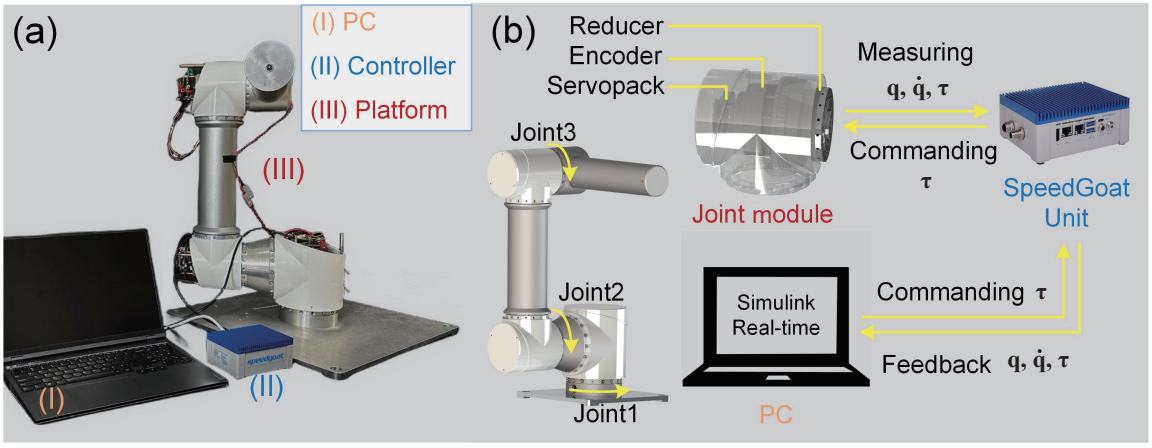

Figure 12 Illustration of the three-axis experimental platform: (a) the experimental setup; (b) the diagram of the information interchange between the main devices.

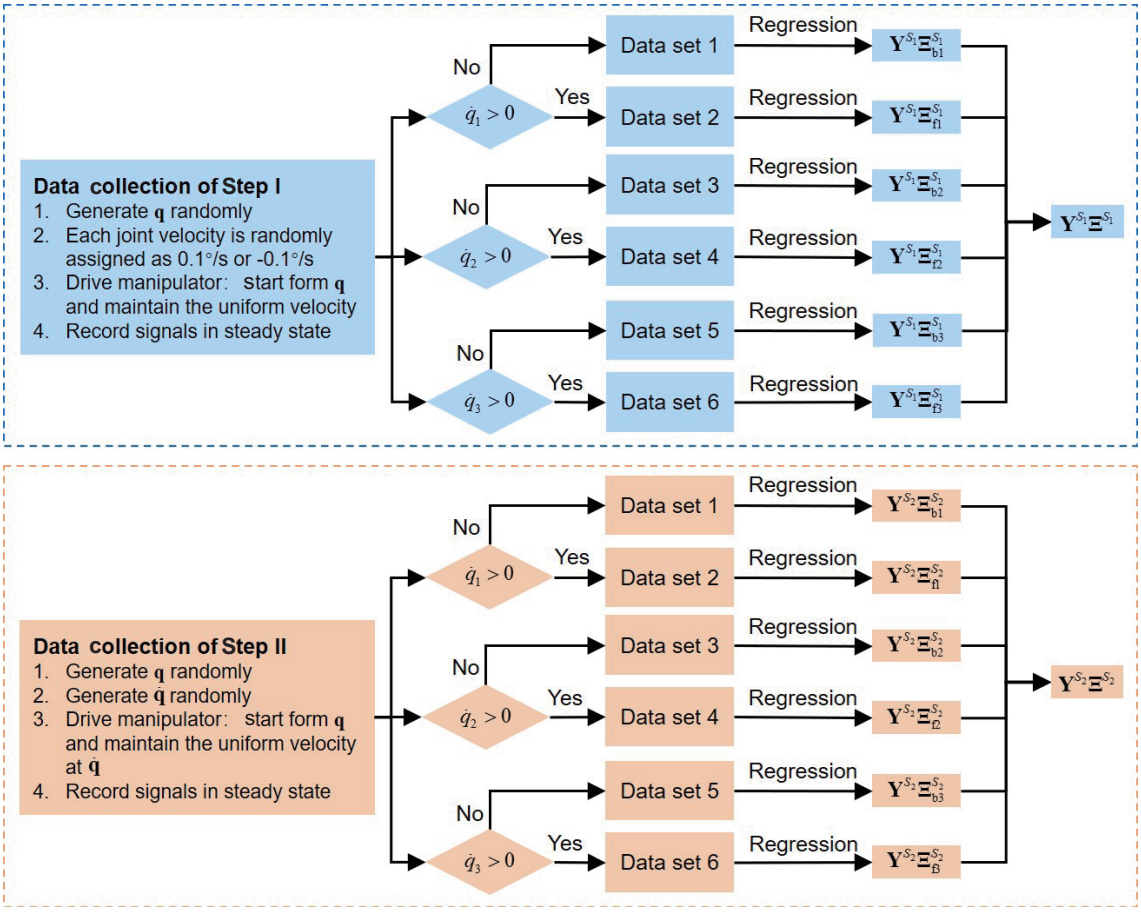

Figure 13 Flow diagram of data collection and model reconstruction for the first two steps of stepwise SINDy in the experiment.

and the results are shown in Fig. 14. The reconstructed model in the first step captures the characteristics of the gravity and the joint dry frictions of the mechanism, so the torque predictions for the first and third joints, being less affected by gravity, are closer to the dry friction. The model reconstructed in the second step further predicts the velocity-dependent dynamics of the mechanism, such as the velocity terms in the friction and the Coriolis and centrifugal terms. The last step reconstructs the complete dynamic model of the three-axis serial manipulator. The experimental results demonstrate that as the number of steps increases, the reconstructed model obtained at each step of the stepwise SINDy becomes more accurate in predicting the joint torques. Additionally, we design the corresponding control strategies based on the reconstructed models obtained from Step I and Step III of stepwise SINDy, respectively given by

$$\boldsymbol{\tau} = K_p\mathbf{e} + K_i\int_0^t \mathbf{e}\mathrm{d}T + K_d\dot{\mathbf{e}} + \hat{\mathbf{Y}}^{S_1}\boldsymbol{\Xi}^{S_1}, \tag{35}$$

where $\mathbf{e}$ is the vector of tracking error of joint angles, and $\hat{\mathbf{Y}}^{S_1}$ is calculated from the reference signals (RS). The controller with the model reconstructed in Step III is designed as

$$\boldsymbol{\tau} = K_p\mathbf{e}+K_i\int_0^t \mathbf{e}\mathrm{d}T+K_d\dot{\mathbf{e}}+\hat{\mathbf{Y}}^{S_1}\boldsymbol{\Xi}^{S_1}+\hat{\mathbf{Y}}^{S_2}\boldsymbol{\Xi}^{S_2}+\hat{\mathbf{Y}}^{S_3}\boldsymbol{\Xi}^{S_3}, \tag{36}$$

where $\hat{\mathbf{Y}}^{S_1}$, $\hat{\mathbf{Y}}^{S_2}$, and $\hat{\mathbf{Y}}^{S_3}$ are also calculated based on the RS. To demonstrate the effectiveness of the proposed model reconstruction method, controllers Eqs. (35) and (36) are to be compared with the classic PID control in the torque mode. The gains of all controllers are identically chosen as $K_p = 80$, $K_i = 8$, and $K_d = 2$. The external trajectory tracking performance of the three controllers is shown in Fig. 15 where the time axis does not represent continuous time. The interval 1-20 s represents stable trajectory tracking under PID control, 20-40 s under the controller in Eq. (35), and 40-60 s under the controller in Eq. (36). The results of the first period of trajectory tracking show that the PID controller has difficulty in overcoming the dry friction during the joint velocity commutation, leading to the overall tracking performance degradation. Denote by $\eta_i = \sum_{i=1}^{n}|e_i|\Big/n$, $i = 1, 2, 3$ the average of absolute tracking errors of three joint angles with unit of deg. Compared to the model-free PID controller ($\eta_1 = 1.10$, $\eta_2 = 1.21$, $\eta_3 = 1.23$), the controller that uses the reconstructed model in Step I of stepwise SINDy to generate the dynamic compensation significantly improves the trajectory tracking accuracy ($\eta_1 = 0.73$, $\eta_2 = 0.50$, $\eta_3 = 0.81$), which is further improved after implementing the reconstructed model obtained from Step III of stepwise SINDy in forming the dynamic

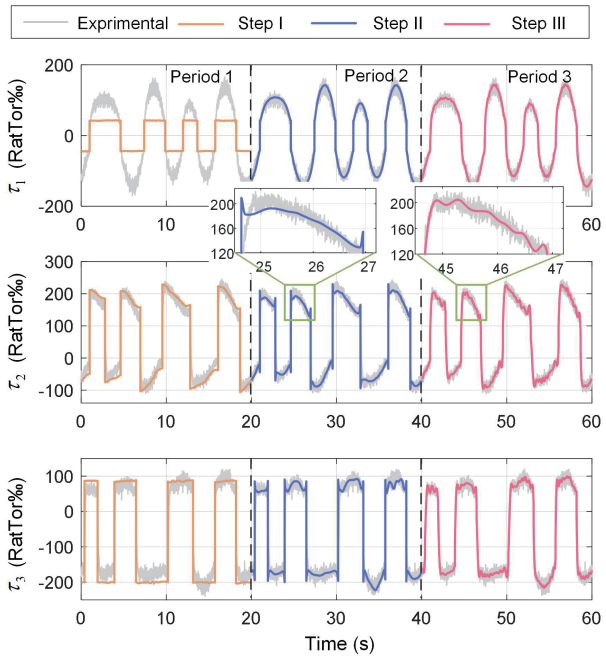

Figure 14 Torque predictions of the reconstructed model obtained in each step of the stepwise SINDy where each reconstructed model predicts the joint torques when tracking a trajectory for one period (20 s) where the trajectory given by $q_1 = k_1\cos\omega_1 t + k_2\sin\omega_2 t + k_3\cos\omega_3 t - k_1 - k_3$, $q_2 = k_1\sin\omega_1 t + k_2\cos\omega_2 t + k_3\sin\omega_3 t - k_2$, $q_3 = k_1\cos\omega_1 t + k_2\cos\omega_2 t + k_3\sin\omega_3 t - k_1 - k_2$, where $k_1 = 2$, $k_2 = 3$, $k_3 = 4$, $\omega_1 = 0.5\pi$, $\omega_2 = 0.4\pi$, and $\omega_3 = 0.3\pi$.

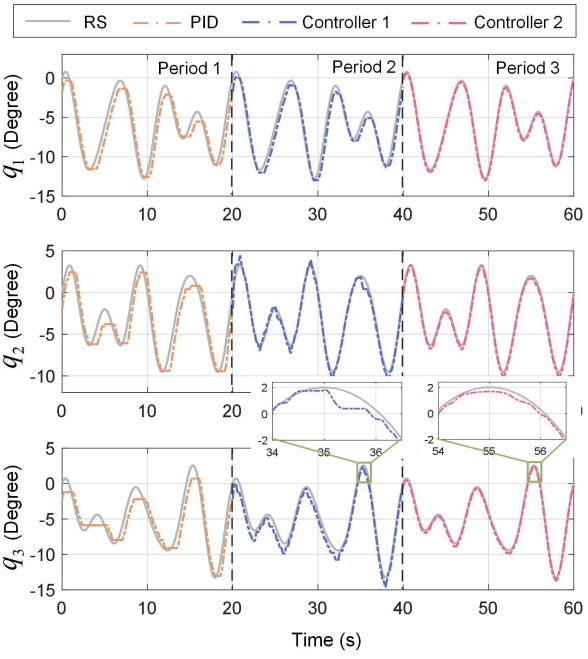

Figure 15 Performance of tracking the external trajectory of the three controllers where the grey curves denote the RS for the joint angles, the yellow, blue, and pink curves denote the time history of the joint angles under the model-free PID controller, the controller based on the reconstructed model obtained from Step I of stepwise SINDy, and the controller based on the reconstructed model obtained from Step III of stepwise SINDy, respectively.

compensation ($\eta_1 = 0.25$, $\eta_2 = 0.25$, and $\eta_3 = 0.27$). It is worth noting that the interpretability of the reconstructed models in the stepwise SINDy supports the design of advanced model-based control laws.

5. Conclusion

This research has proposed a stepwise data-driven approach for the dynamic model reconstruction of multi-DoF serial manipulators. The primary contributions of this study are summarized as follows:

(1) Clustering of data features: A relative activation indicator is proposed to classify the data features of multi-DoF serial manipulators, revealing the physical meaning of each cluster. This classification forms the basis for the stepwise SINDy method.

(2) Proposal of double-feature-set library generation: To improve the SINDy-based symbolic regression, DLG is proposed to reduce unnecessary candidate functions and generalize across serial manipulators with common joint types, such as revolute and prismatic joints.

(3) Method validation: Simulations on a 4-DoF SCARA manipulator and a three-axis serial manipulator show that stepwise SINDy reduces library size and improves computational efficiency. In noisy data conditions, the reconstructed model shows better sparsity and predictive accuracy. Experimental validation on a three-axis platform, considering nonlinear joint frictions, demonstrates accurate torque predictions and promising trajectory tracking, highlighting potential for practical applications of the method.

In conclusion, the contributions of this research include the development of RAI-based data clustering, the proposed stepwise SINDy approach, and the proposal of DLG, all of which address the limitations of existing techniques and demonstrate improvements in dynamic model reconstruction of multi-DoF serial manipulators. There are two perspectives that require further refinement in the future work. Firstly, an exploration of the potential applications of data mining algorithms in the dynamic modeling of robotic manipulators will be pursued. For instance, utilizing data noise and correlation as parameters for unsupervised learning offers innovative possibilities for data-driven model reconstruction. Secondly, addressing dimensionality challenges in model reconstruction for industrial robotic manipulators remains critical. Strategies will be devised to address high-dimensional system complexities in a more concise and efficient manner.

Conflict of interest *On behalf of all authors, the corresponding author states that there is no conflict of interest.*

Author contributions ***Dingxu Guo:*** *Formal analysis, Investigation, Software, Visualization, Methodology, Validation, Writing – original draft.* ***Jian Xu:*** *Conceptualization, Resources, Supervision, Funding acquisition, Project administration.* ***Xiaoxu Zhang:*** *Conceptualization, Formal analysis, Resources, Visualization, Supervision, Funding acquisition, Project administration.* ***Xiuting Sun:*** *Conceptualization, Supervision, Funding acquisition, Project administration.* ***Shu Zhang:*** *Conceptualization, Investigation, Methodology, Validation, Funding acquisition, Supervision, Writing – review & editing.*

Acknowledgements *This work was supported by the National Natural Science Foundation of China (Grant Nos. 12072237, 12472022, 12372022, 12372065, and U2441202) and the Fundamental Research Funds for the Central Universities (Grant No. 22120220590).*

1 R. Li, and H. Qiao, A survey of methods and strategies for high-precision robotic grasping and assembly tasks—Some new trends, IEEE ASME Trans. Mech. **24**, 2718 (2019).
2 V. L. Nguyen, C. H. Kuo, and P. T. Lin, Performance analysis of gravity-balanced serial robotic manipulators under dynamic loads, Mech. Mach. Theory **191**, 105519 (2024).
3 S. Pedrammehr, S. Nahavandi, and H. Abdi, Closed-form dynamics of a hexarot parallel manipulator by means of the principle of virtual work, Acta Mech. Sin. **34**, 883 (2018).
4 C. Ye, J. Yang, and H. Ding, High-accuracy prediction and compensation of industrial robot stiffness deformation, Int. J. Mech. Sci. **233**, 107638 (2022).
5 C. Della Santina, C. Duriez, and D. Rus, Model-based control of soft robots: A survey of the state of the art and open challenges, IEEE Control Syst. **43**, 30 (2023).
6 M. Indri, and S. Trapani, Framework for static and dynamic friction identification for industrial manipulators, IEEE ASME Trans. Mechatron. **25**, 1589 (2020).
7 D. Tan, J. Zhou, K. Wang, C. Cai, and D. Xu, Modeling and analysis of the friction in a non-linear sliding-mode triboelectric energy harvester, Acta Mech. Sin. **38**, 521330 (2022).
8 Y. H. Jia, Q. Hu, and S. J. Xu, Dynamics and adaptive control of a dual-arm space robot with closed-loop constraints and uncertain inertial parameters, Acta Mech. Sin. **30**, 112 (2014).
9 Z. Li, S. Li, and X. Luo, An overview of calibration technology of industrial robots, IEEE CAA J. Autom. Sin. **8**, 23 (2021).
10 Y. Yuan, and W. Sun, An integrated kinematic calibration and dynamic identification method with only static measurements for serial robot, IEEE ASME Trans. Mech. **28**, 2762 (2023).
11 K. J. Park, Fourier-based optimal excitation trajectories for the dynamic identification of robots, Robotica **24**, 625 (2006).
12 T. Lee, B. D. Lee, and F. C. Park, Optimal excitation trajectories for mechanical systems identification, Automatica **131**, 109773 (2021).
13 M. Gautier, and W. Khalil, Exciting trajectories for the identification of base inertial parameters of robots, Int. J. Robotics Res. **11**, 362 (1992).
14 J. Wu, J. Wang, and Z. You, An overview of dynamic parameter identification of robots, Robotics Comput.-Int. Manuf. **26**, 414 (2010).
15 C. H. Menq, J. H. Borm, and J. Z. Lai, Identification and observability measure of a basis set of error parameters in robot calibration, J. Mech. Trans. Autom. Des. **111**, 513 (1989).
16 T. Wang, M. Noori, W. A. Altabey, Z. Wu, R. Ghiasi, S. C. Kuok, A. Silik, N. S. D. Farhan, V. Sarhosis, and E. N. Farsangi, From model-driven to data-driven: A review of hysteresis modeling in structural and mechanical systems, Mech. Syst. Signal Process. **204**, 110785 (2023).
17 J. Qian, X. Sun, and J. Xu, A data-driven reconstruction method for dynamic systems with multistable property, NOnlinear Dyn. **111**, 4517 (2023).
18 W. Bradley, J. Kim, Z. Kilwein, L. Blakely, M. Eydenberg, J. Jalvin, C. Laird, and F. Boukouvala, Perspectives on the integration between first-principles and data-driven modeling, Comput. Chem. Eng. **166**, 107898 (2022).

19 Y. Chen, M. Guo, Z. Chen, Z. Chen, and Y. Ji, Physical energy and data-driven models in building energy prediction: A review, Energy Rep. **8**, 2656 (2022).
20 M. Ghasemi, M. Samadi, E. Soleimanian, and K. W. Chau, A comparative study of black-box and white-box data-driven methods to predict landfill leachate permeability, Environ. Monit. Assess. **195**, 862 (2023).
21 J. P. Panda, and H. V. Warrior, Evaluation of machine learning algorithms for predictive Reynolds stress transport modeling, Acta Mech. Sin. **38**, 321544 (2022).
22 Z. Yuan, Y. Wang, C. Xie, and J. Wang, Deconvolutional artificial-neural-network framework for subfilter-scale models of compressible turbulence, Acta Mech. Sin. **37**, 1773 (2021).
23 F. L. Fan, J. Xiong, M. Li, and G. Wang, On interpretability of artificial neural networks: A survey, IEEE Trans. Radiat. Plasma Med. Sci. **5**, 741 (2021).
24 L. S. Coelho, and M. W. Pessôa, Nonlinear model identification of an experimental ball-and-tube system using a genetic programming approach, Mech. Syst. Signal Process. **23**, 1434 (2009).
25 H. Xie, Y. Zhao, and Y. Zhang, Data-driven nonlinear K-L turbulent mixing model via gene expression programming method, Acta Mech. Sin. **39**, 322315 (2023).
26 M. Schmidt, and H. Lipson, Distilling free-form natural laws from experimental data, Science **324**, 81 (2009).
27 Z. Fan, Z. Wang, W. Li, X. Zhu, B. Hu, A. M. Zou, W. Bao, M. Gu, Z. Hao, and Y. Jin, Automated pattern generation for swarm robots using constrained multi-objective genetic programming, Swarm Evol. Comput. **81**, 101337 (2023).
28 F. Chapelle, and P. Bidaud, Closed form solutions for inverse kinematics approximation of general 6R manipulators, Mech. Mach. Theory **39**, 323 (2004).
29 Z. Zhang, and Z. Chen, Modeling and control of robotic manipulators based on symbolic regression, IEEE Trans. Neural Netw. Learn. Syst. **34**, 2440 (2023).
30 K. Danai, and W. G. La Cava, Controller design by symbolic regression, Mech. Syst. Signal Process. **151**, 107348 (2021).
31 S. L. Brunton, J. L. Proctor, and J. N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proc. Natl. Acad. Sci. U.S.A. **113**, 3932 (2016), arXiv: 1509.03580.
32 Z. Huang, Y. Tian, C. Li, G. Lin, L. Wu, Y. Wang, and H. Jiang, Data-driven automated discovery of variational laws hidden in physical systems, J. Mech. Phys. Solids **137**, 103871 (2020).
33 H. K. Chu, and M. Hayashibe, Discovering interpretable dynamics by sparsity promotion on energy and the lagrangian, IEEE Robot. Autom. Lett. **5**, 2154 (2020).
34 S. Askarinejad, A. Fahim, M. Yazdi, and M. T. Masouleh, in Data-driven identification of the Jacobian matrix of a 2-DoF spherical parallel manipulator: Proceedings of 2019 7th International Conference on Robotics and Mechatronics (ICRoM), Tehran, 2019, pp. 229-234.
35 C. Yu, Z. Li, D. Yang, and H. Liu, A fast robotic arm gravity compensation updating approach for industrial application using sparse selection and reconstruction, Robot. Auton. Syst. **149**, 103971 (2022).
36 K. Kaheman, J. N. Kutz, and S. L. Brunton, SINDy-PI: A robust algorithm for parallel implicit sparse identification of nonlinear dynamics, Proc. R. Soc. A. **476**, 20200279 (2020), arXiv: 2004.02322.
37 M. Omar, R. Li, and A. Asker, A framework for data driven dynamic modeling of serial manipulators, IEEE Access **10**, 124874 (2022).
38 G. J. Oyewole, and G. A. Thopil, Data clustering: Application and trends, Artif. Intell. Rev. **56**, 6439 (2023).
39 A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data, Inf. Sci. **622**, 178 (2023).
40 A. Saxena, M. Prasad, A. Gupta, N. Bharill, O. P. Patel, A. Tiwari, M. J. Er, W. Ding, and C. T. Lin, A review of clustering techniques and developments, Neurocomputing **267**, 664 (2017).
41 D. Fernex, B. R. Noack, and R. Semaan, Cluster-based network modeling-From snapshots to complex dynamical systems, Sci. Adv. **7**, eabf5006 (2021).
42 J. J. Bramburger, D. Dylewsky, and J. N. Kutz, Sparse identification of slow timescale dynamics, Phys. Rev. E **102**, 022204 (2020), arXiv: 2006.00940.
43 J. Slotine, and W. Li, Adaptive manipulator control: A case study, IEEE Trans. Automat. Contr. **33**, 995 (1988).
44 M. Cui, Introduction to the K-means clustering algorithm based on the elbow method, Account. Audit. Fina. **1**, 5 (2020).
45 M. A. Rasmussen, and R. Bro, A tutorial on the Lasso approach to sparse modeling, Chemometr. Intell. Lab. Syst. **119**, 21 (2012).
46 A. Visioli, and G. Legnani, On the trajectory tracking control of industrial SCARA robot manipulators, IEEE Trans. Ind. Electron. **49**, 224 (2002).
47 Y. Yan, J. Xu, and M. Wiercigroch, Modelling of regenerative and frictional cutting dynamics, Int. J. Mech. Sci. **156**, 86 (2019).

串联机械臂模型重构: 一种分步数据驱动方法

郭丁旭, 徐鉴, 张晓旭, 孙秀婷, 张舒

摘要 机械臂动力学建模方法的发展对于有效部署基于模型的控制至关重要. 传统方法依赖于基于第一原理的动力学建模和精确的参数识别, 而本文则通过数据驱动的模型重构探索了另一种方法. 为解决多自由度串联机械臂模型重构中的维度灾难问题, 本文提出了一种相对激活指标, 并在此基础上使用k-means聚类算法对不同工况下机械臂的数据进行分类. 随后, 我们利用基础的先验知识找到每个聚类的动力学特征, 并使用非线性动力学稀疏识别(SINDy)方法分步重构动力学模型. 为生成SINDy候选函数库, 本文提出了针对具有常见关节类型的串联机械臂的双特征集策略. 仿真结果表明, 分步模型重构方法不仅减少了候选函数库的规模, 还降低了数据噪声对重构结果的影响. 最后, 我们在实验平台上部署了基于重构模型的控制器, 实验结果证明了该方法在实验平台轨迹跟踪性能方面的提升以及在工程应用方面的潜力.