Mutation detection and fast identification of switching system based on data-driven method

Zhonghua Zhang(张钟化), Wei Xu(徐伟)†, and Yi Song(宋怡)

School of Mathematics and Statistics, Northwestern Polytechnical University, Xi'an 710072, *China*

(Received 24 October 2022; revised manuscript received 5 December 2022; accepted manuscript online 8 February 2023)

In the engineering field, switching systems have been extensively studied, where sudden changes of parameter value and structural form have a significant impact on the operational performance of the system. Therefore, it is important to predict the behavior of the switching system, which includes the accurate detection of mutation points and rapid reidentification of the model. However, few efforts have been contributed to accurately locating the mutation points. In this paper, we propose a new measure of mutation detection — the threshold-based switching index by analogy with the Lyapunov exponent. We give the algorithm for selecting the optimal threshold, which greatly reduces the additional data collection and the relative error of mutation detection. In the system identification part, considering the small data amount available and noise in the data, the abrupt sparse Bayesian regression (abrupt-SBR) method is proposed. This method captures the model changes by updating the previously identified model, which requires less data and is more robust to noise than identifying the new model from scratch. With two representative dynamical systems, we illustrate the application and effectiveness of the proposed methods. Our research contributes to the accurate prediction and possible control of switching system behavior.

Keywords: mutation detection, switching index, system identification, sparse Bayesian regression

PACS: 02.30.Hq, 05.45.–a, 02.50.–r, 02.60.Pn **DOI:** 10.1088/1674-1056/acb9f5

1. Introduction

With the rapid development of computer, military, biology, and artificial intelligence, many systems in modern industry are becoming more and more complex.[1–4] Some systems have sudden changes in parameter values or structural forms.[4] For example, a variable-wing aircraft actively switches wing configurations to perform different flight missions.[5] A machinery exposed to extreme environment passively alters the damping and stiffness of the system. The above systems whose parameter values and structural forms change with time are called switching systems. In this paper, we define the switching system as follows: a system in which parameters and structural forms change abruptly caused by external forcing or interactions between systems at any time (*e.g.*, the change of parameter values, addition and deletion of an item of the system). The response, reliability, and stability of switching systems vary greatly with time. Therefore, it is crucial to master the real-time control equations of the system, which helps predict the dynamics and operational performance of the system.[6]

Due to the complexity of the system and the lack of scientific understanding of the system, it is usually difficult to derive the control equations of the system based on the first principles such as conservation law and physical principle. Fortunately, with the advancement of sensor technology, there are more technical means to measure the data of the system. Therefore, it is desirable to model the switching system from the data-driven perspective. Traditional data-driven approaches are not currently applicable for switching systems because they assume that the data collected comes from the same set of equations. However, due to the changes of switching systems with time, the data collected would come from different equations if the exact location of the switching point is not known. Thus, the accurate location of the switching points plays a crucial role in identifying the switching systems.

With regard to the mutation detection in systems, the current research mainly focuses on using some information-theoretic measures to characterize the complexity of time series, so as to judge the mutation of the system. We know that when the system switches, the probability density distribution of system variables will change greatly. The information-theoretic measure is a function of the probability density distribution of the system variables, so when the parameter value or structure of the system changes, the information-theoretic measure will change considerably. Traditional detection methods such as sliding t-test, Cramer method, Yamamoto signal-to-noise ratio and Mann-Kendall method have been widely used in the early stage of mutation detection,[7–9] but there are different degrees of mutation point drift in practical detection. In this situation, a series of new methods have been successively developed. Based on the scaling theory, detrended fluctuation analysis and sliding detrended fluctuation analysis have been developed in succession,[10] which can detect the dynamic structural mutation for actual observation data. However, the prerequisite for applying these methods is that the time series analyzed should satisfy the fractal characteris-

†Corresponding author. E-mail: weixunpu@nwpu.edu.cn

© 2023 Chinese Physical Society and IOP Publishing Ltd

tics, but not all time series have this property. Approximate entropy and dynamic correlation factor index have the ability to distinguish different structures of system,[11,12] while they usually can not pinpoint the location of mutation points accurately, only give a rough mutation region. All these studies have greatly enriched the mutation detection methods and broadened the avenues of mutation detection. However, due to the complexity of the problems studied, there cannot be a universal method to solve all problems once and for all. Thus, how to accurately detect the mutation points of the system needs to be further explored.

Due to the random variation of switching systems, only a small amount of data is available, and the time used to identify system is relatively short. A large number of techniques have been used to characterize the system dynamics from data.[13,14] Classical linear model identification algorithms include the Kalman filter,[15–17] eigensystem realization algorithm (ERA),[18] and dynamic mode decomposition (DMD).[19–22] The resulting linear models from these methods are excellent for model predictive control, but could not derive expressions describing the behavior of the system. Machine learning is also increasingly being used for the identification of nonlinear dynamical systems. Neural network has been utilized for system identification for decade.[14,23–27] It shows a good dynamic recovery ability, but the large amount of training time and data is unrealistic, and the mechanism of the system is not well revealed, thus, could not be used for rapid identification of switching systems.

Recently, the sparse regression method has been widely applied in the identification of various nonlinear dynamical systems,[28–37] which can provide explicit formulas of the differential equations, making further analysis on the systems possible. This method has been extended to the fast recovery of nonlinear dynamical systems with switching parameters and structures.[36] A data-driven approach established under the variational framework has also been generalized to the identification and re-identification of switching systems.[38,39]

The aforementioned work on the identification of switching system adopt the threshold-based method to detect the occurrence of switching and identify the dynamic equation quickly and accurately after switching, but without discussing the selection of threshold in detail. In Ref. [36], the mutation of system is judged by the inconsistency of the time scale given by the local Lyapunov exponent and the prediction horizon. In this paper, analogizing with the definition of Lyapunov exponent, we give the concept of the threshold-based switching index to detect system switching. The methods for detecting mutations in the reference and the one that will be presented below all rely on the selection of thresholds. The difference is that the threshold in the literature is fixed, while in this paper, we give an algorithm to select the optimal threshold, which can reduce the unnecessary data acquisition and relative error in switching detection. The results of numerical experiments show that the switching point detected with the switching index is very close to the real switching point, which is beneficial for realtime tracking of system changes. For the identification of the system after switching, the sparse Bayesian regression (SBR) is adopted[40] and combined with the abrupt-SINDy framework to update existing models with sparsity correction in this paper,[36] including the changes of parameter values, deletions, and additions of terms. The results indicate that, compared with the traditional SBR method, the abrupt-SBR method has the advantages of less required data and more robust to noise, enabling a fast and accurate update of the model after the sudden switching of the system.

The rest of the paper is organized as follows. In Section 2, the process of sparse Bayesian regression for identifying dynamical systems and abrupt-SBR framework are introduced. In Section 3, the concept of the threshold-based switching index is introduced, and the algorithm for selecting the optimal threshold is presented. In Section 4, two representative switching systems, *i.e.*, the switching Van der Pol system and Duffing oscillator with parameter changes, are investigated to verify the effectiveness of the threshold-based switching index and abrupt-SBR. Finally, the conclusions are summarized in Section 5.

2. Sparse Bayesian regression and abrupt-SBR framework for identifying switching systems

2.1. Sparse Bayesian regression method

Most physical systems have only a few important terms governing the dynamics. The governing equations are sparse in the high-dimensional nonlinear function space. Recently, sparse regression in the candidate nonlinear function library has been widely used in the sparse identification of nonlinear dynamics (SINDy) to effectively identify the governing equations of the system from the data. Consider the dynamical system form

$$\dot{\boldsymbol{x}}(t) = \frac{\mathrm{d}\boldsymbol{x}(t)}{\mathrm{d}t} = f(\boldsymbol{x}(t)), \tag{1}$$

where $x(t) \in \mathrm{R}^n$ denotes the state of system at time t, $f(\boldsymbol{x}(t))$ represents the dynamic constraints that consists of only a few terms. According to SINDy, the system can be represented by the data matrixes as[28]

$$\dot{\boldsymbol{X}} = \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{\Xi}, \tag{2}$$

where $\boldsymbol{\Theta}(\boldsymbol{X})$ is the candidate function library on the right-hand side of Eq. (1), $\boldsymbol{\Xi} = [\xi_1 \xi_2 \cdots \xi_n]$ is the sparse vectors of coefficients. Each column ξ_k of $\boldsymbol{\Xi}$ determines which terms on the right-hand side of the k-th row $\dot{\boldsymbol{x}}_k = f_k(\boldsymbol{x})$ $(k = 1, 2, \ldots, n)$ of Eq. (1) are active. In this paper, the sparse Bayesian regression method is taken to get the sparse matrix $\boldsymbol{\Xi}$.

In the Bayesian approach, it is recognized that the time series data matrix should be written explicitly as[40]

$$\dot{\boldsymbol{X}} = \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{\Xi} + \varepsilon, \tag{3}$$

instead of Eq. (2), where ε is a noise or error term, representing the uncertainty in data and models.

We consider the k-th row $\dot{\boldsymbol{x}}_k = f_k(\boldsymbol{x})$ $(k = 1, 2, \ldots, n)$ of Eq. (1), then the unknown coefficient ξ_k can be obtained by solving the following equation:

$$\dot{\boldsymbol{X}}_k = \boldsymbol{\Theta}(\boldsymbol{X})\xi_k + \varepsilon_k, \tag{4}$$

where $\dot{\boldsymbol{X}}_k$ represents the k-th column of $\dot{\boldsymbol{X}}$. The Bayesian framework assumes the error ε_k obeys a Gaussian distribution with mean zero and variance σ^2. Meanwhile, they are independent and identically distributed for each k.

The Bayesian model gives the multivariate Gaussian likelihood function for the vector ξ_k

$$p(\dot{\boldsymbol{X}}_k|\xi_k, \sigma^2) = (2\pi\sigma^2)^{-N/2} \exp\left\{-\frac{\left\|\dot{\boldsymbol{X}}_k - \boldsymbol{\Theta}(\boldsymbol{X})\xi_k\right\|}{2\sigma^2}\right\}, \tag{5}$$

where N is the number of data in the sample. Suppose the prior distribution of the unknown vector ξ_k is as follows:

$$p(\xi_k|\alpha) = \prod_{j=1}^{M} \boldsymbol{N}(\xi_{kj}|0, \alpha_j^{-1}), \tag{6}$$

where $\alpha = [\alpha_1, \ldots, \alpha_M]^{\mathrm{T}}$. Each α_j controls each ξ_{kj} individually. This form of prior distribution promotes model sparsity.[41]

The goal of the sparse Bayesian method is to obtain the posterior probability of ξ_k based on the assumption of ξ_k prior distribution at the given observational data of $\dot{\boldsymbol{X}}_k$, that is $p(\xi_k|\dot{\boldsymbol{X}}_k)$.

According to the multiplicative probability formula, the posterior probability of all unknown parameters given the data $\dot{\boldsymbol{X}}_k$ can be decomposed into the following form:

$$p(\xi_k, \alpha, \sigma^2|\dot{\boldsymbol{X}}_k) = p(\xi_k|\dot{\boldsymbol{X}}_k, \alpha, \sigma^2)p(\alpha, \sigma^2|\dot{\boldsymbol{X}}_k). \tag{7}$$

We use Dirac function at $(\hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}})$ to approximate $p(\alpha, \sigma^2|\dot{\boldsymbol{X}}_k)$, then,

$$p(\xi_k, \alpha, \sigma^2|\dot{\boldsymbol{X}}_k) \approx p(\xi_k|\dot{\boldsymbol{X}}_k, \alpha, \sigma^2)\delta(\hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}}), \tag{8}$$

where $\hat{\alpha}_{\mathrm{ML}}$ and $\hat{\sigma}^2_{\mathrm{ML}}$ are the maximum likelihood estimators of α and σ^2, respectively. We do so on the basis that the likelihood function $p(\alpha, \sigma^2|\dot{\boldsymbol{X}}_k)$ takes the largest value at the maximum likelihood estimate $\hat{\alpha}_{\mathrm{ML}}$,$\hat{\sigma}^2_{\mathrm{ML}}$ and the smaller value at other parameter values. We assume that the value of the likelihood function $p(\alpha, \sigma^2|\dot{\boldsymbol{X}}_k)$ is concentrated at $\hat{\alpha}_{\mathrm{ML}}$,$\hat{\sigma}^2_{\mathrm{ML}}$, which is similar to the definition of the Dirac function

$$\begin{aligned} &\delta(\alpha, \sigma^2) = \delta(\alpha - \hat{\alpha}_{\mathrm{ML}}, \sigma^2 - \hat{\sigma}^2_{\mathrm{ML}}) = 0, \\ &(\alpha \neq \hat{\alpha}_{\mathrm{ML}}, \quad \sigma^2 \neq \hat{\sigma}^2_{\mathrm{ML}}). \end{aligned} \tag{9}$$

Also,

$$\begin{aligned} p(\alpha, \sigma^2|\dot{\boldsymbol{X}}_k) &= \frac{p(\dot{\boldsymbol{X}}_k|\alpha, \sigma^2)p(\alpha, \sigma^2)}{p(\dot{\boldsymbol{X}}_k)} \\ &= \frac{p(\dot{\boldsymbol{X}}_k|\alpha, \sigma^2)p(\alpha)p(\sigma^2)}{p(\dot{\boldsymbol{X}}_k)}, \end{aligned} \tag{10}$$

where $p(\dot{\boldsymbol{X}}_k)$, $p(\alpha)$, and $p(\sigma^2)$ are constants. Therefore, $p(\alpha, \sigma^2|\dot{\boldsymbol{X}}_k) \propto p(\dot{\boldsymbol{X}}_k|\alpha, \sigma^2)$. Then the derivation of the maximum likelihood estimate $\hat{\alpha}_{\mathrm{ML}}$ and $\hat{\sigma}^2_{\mathrm{ML}}$ is as follows:

$$\begin{aligned} (\hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}}) &= \arg\max_{\alpha,\sigma^2}\{p(\alpha, \sigma^2|\dot{\boldsymbol{X}}_k)\} \Leftrightarrow \arg\max_{\alpha,\sigma^2}\{p(\dot{\boldsymbol{X}}_k|\alpha, \sigma^2)\} = \arg\max_{\alpha,\sigma^2} \int p(\dot{\boldsymbol{X}}_k, \xi_k|\alpha, \sigma^2)\,\mathrm{d}\xi_k \\ &= \arg\max_{\alpha,\sigma^2} \int p(\dot{\boldsymbol{X}}_k|\xi_k, \alpha, \sigma^2)p(\xi_k|\alpha, \sigma^2)\,\mathrm{d}\xi_k = \arg\max_{\alpha,\sigma^2} \int p(\dot{\boldsymbol{X}}_k|\xi_k, \sigma^2)p(\xi_k|\alpha)\,\mathrm{d}\xi_k \\ &= \arg\max_{\alpha,\sigma^2} \left\{(2\pi)^{-N/2}|\sigma^2\boldsymbol{I} + \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{A}^{-1}(\boldsymbol{\Theta}(\boldsymbol{X}))^{\mathrm{T}}|^{-1/2} \exp\left\{-\frac{1}{2}\xi_k^{\mathrm{T}}(\sigma^2\boldsymbol{I} + \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{A}^{-1}(\boldsymbol{\Theta}(\boldsymbol{X}))^{\mathrm{T}})^{-1}\xi_k\right\}\right\}, \end{aligned} \tag{11}$$

where $\boldsymbol{A} = \mathrm{diag}(\alpha_1, \ldots, \alpha_M)$. This maximization is called the type-2 maximum-likelihood. There is currently no analytical solution to this maximization. In this paper, we adopt the fast sequence sparse Bayesian learning algorithm for numerical solutions.[42]

After obtaining the above formulas, we can integrate α and σ^2 to get the posterior probability of the vector ξ_k under the given data $\dot{\boldsymbol{X}}_k$ condition:

$$\begin{aligned} p(\xi_k|\dot{\boldsymbol{X}}_k) &= \iint p(\xi_k, \alpha, \sigma^2|\dot{\boldsymbol{X}}_k)\,\mathrm{d}\alpha\,\mathrm{d}\sigma^2 = \iint p(\xi_k|\dot{\boldsymbol{X}}_k, \alpha, \sigma^2)\delta(\hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}})\,\mathrm{d}\alpha\,\mathrm{d}\sigma^2 = p(\xi_k|\dot{\boldsymbol{X}}_k, \hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}}) \\ &= \frac{p\left(\dot{\boldsymbol{X}}_k, \xi_k|\hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}}\right)}{p(\dot{\boldsymbol{X}}_k|\hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}})} = \frac{p\left(\dot{\boldsymbol{X}}_k|\xi_k, \hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}}\right)p\left(\xi_k|\hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}}\right)}{p(\dot{\boldsymbol{X}}_k|\hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}})} = \frac{p\left(\dot{\boldsymbol{X}}_k|\xi_k, \hat{\sigma}^2_{\mathrm{ML}}\right)p\left(\xi_k|\hat{\alpha}_{\mathrm{ML}}\right)}{p(\dot{\boldsymbol{X}}_k|\hat{\alpha}_{\mathrm{ML}}, \hat{\sigma}^2_{\mathrm{ML}})} \\ &= (2\pi)^{-M/2}|\hat{\boldsymbol{\Sigma}}|^{-1/2} \exp\left\{-\frac{1}{2}(\xi_k - \hat{\mu})^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}^{-1}(\xi_k - \hat{\mu})\right\}, \end{aligned} \tag{12}$$

where the posterior variance and mean are as follows:

$$\begin{cases} \hat{\boldsymbol{\Sigma}} = \left[\hat{\sigma}_{ml}^{-2}(\boldsymbol{\Theta}(\boldsymbol{X}))^{\mathrm{T}}\boldsymbol{\Theta}(\boldsymbol{X}) + \mathrm{diag}(\hat{\alpha}_{ml})\right]^{-1}, \\ \hat{\mu} = \hat{\sigma}_{ml}^{-2}\,\hat{\boldsymbol{\Sigma}}\,(\boldsymbol{\Theta}(\boldsymbol{X}))^{\mathrm{T}}\dot{\boldsymbol{X}}_k. \end{cases} \tag{13}$$

The posterior distribution of each component of the coefficient vector ξ_k is

$$p(\xi_{kj}|\dot{\boldsymbol{X}}_k) = \mathrm{N}(\xi_{kj}|\hat{\mu}_j, \hat{\boldsymbol{\Sigma}}_{jj}), \tag{14}$$

where $\hat{\mu}_j$ is the j-th element of vector $\hat{\mu}$ and $\hat{\boldsymbol{\Sigma}}_{jj}$ is the j-th element on the diagonal of matrix $\hat{\boldsymbol{\Sigma}}$. Same as the sequence threshold least squares method,[28] a threshold γ is introduced to improve accuracy and sparsity. Set to zero for any coefficient ξ_{kj} less than the threshold. Then the sparse Bayesian regression process is iterated repeatedly on the non-zero term of ξ_k until the algorithm converges. Algorithm 1 shows the specific iterative steps.

Algorithm 1 Iterative process of sparse Bayesian regression

1 **Input:** $\dot{\boldsymbol{X}}_k$, $\boldsymbol{\Theta}(\boldsymbol{X})$, threshold γ.
2 Calculate $p(\xi_k|\dot{\boldsymbol{X}}_k)$, where the mean is $\hat{\mu}$. Set the component of $\hat{\mu}$ whose absolute value is less than γ to zero.
while $\hat{\mu} \neq 0$ **do**
Delete the columns with the corresponding weight zero of $\boldsymbol{\Theta}(\boldsymbol{X})$, denoting $\boldsymbol{\Theta}'(\boldsymbol{X})$;
Calculate $p(\xi_k'|\dot{\boldsymbol{X}}_k)$ in $\dot{\boldsymbol{X}}_k = \boldsymbol{\Theta}'(\boldsymbol{X})\xi_k'$, where the mean is $\hat{\mu}'$, the posterior covariance is $\hat{\boldsymbol{\Sigma}}$;
$\hat{\mu} = \hat{\mu}'$;
Set the component of $\hat{\mu}$ whose absolute value is less than γ to zero.
if $\hat{\mu}$ is the same as the previous loop, **then**
break;
end
end
$\hat{\boldsymbol{\Sigma}}$ is the posterior covariance obtained from the last loop.
3 **Output:** $\hat{\mu}$, $\hat{\boldsymbol{\Sigma}}$.

2.2. Abrupt-SBR framework

We have defined that mutation of the switching system involves only changes of parameter values, additions or deletions of terms. It indicates that we can perform sparse Bayesian regression on the existing sparse structure $\boldsymbol{\Theta}_0$ to identify the certain kind of change. The existing sparse structure here refers to the basis function composition of the system before switching. This process uses less data and enables faster and more accurate identification of governing equations than identifying new models from scratch, which is undoubtedly beneficial for the identification of the system. We call the process of sparse correction of existing models using sparse Bayesian regression as abrupt-SBR, which mainly handles the following three basic types of model changes.

(i) Changes of parameter value and deletion of terms

We perform the sparse Bayesian regression process on the existing sparse structure $\boldsymbol{\Theta}_0$, and if the error between the observation response and prediction response is within the allowable range, it means that the system has a change in terms or a reduction in terms. In this way, we get new parameter values or identify which terms have dropped out, which is fast in computation and requires a small amount of data. If there is a large residual error, exploring the addition of terms is necessary.

(ii) Addition of terms

If the mutation of the switching system involves only the addition of terms, then based on the residual error, we perform the sparse Bayesian identification process on the remaining inactive terms to find the minimum number of terms that can account for the error. The remaining inactive terms here refer to the removal of elements of the existing sparse structure $\boldsymbol{\Theta}_0$ from the basis function library $\boldsymbol{\Theta}$.

3. The detection of model mutation

In this section, we propose a concept of the threshold-based switching index to detect system mutation. A series of thresholds are generated by dichotomous successive approximation algorithm, and the effect of these thresholds on mutation detection is analyzed to select an optimal threshold.

3.1. Definition of the threshold-based switching index

We know the Lyapunov exponent is an important indicator of the system dynamics. It characterizes the average exponential rate of convergence or divergence between adjacent orbits of the system in phase space. Whether the system is chaotic can be intuitively judged by whether the maximum Lyapunov exponent is greater than zero. The maximum Lyapunov exponent greater than zero indicates the system is chaotic.

In this section, the system we consider is Eq. (1), whose Lyapunov exponent is expressed as follows:[43]

$$\lambda(\xi(t)), \xi(0)) = \lim_{t\to\infty} \lim_{\xi(0)\to 0} \frac{\log\|\xi(t)\| - \log\|\xi(0)\|}{t}, \tag{15}$$

where $\xi(0)$ represents the distance between the two orbits at the initial moment, and $\xi(t)$ represents the distance between the two orbits at time t as time evolves.

Next, we give the concept of switching index analogizing with the Lyapunov exponent. We determine whether the system switches or not according to whether the switching index is greater than zero. Below we give the expression of the switching index

$$\lambda'(t) = \log(\Delta x(t)) - \log(\Delta x), \tag{16}$$

where

$$\Delta x(t) = \frac{\sum_{j=1}^{N} \sum_{i(t_i\in[0,t])} |\hat{x}_j(t_i) - x_j(t_i)|}{\sum_{i(t_i\in[0,t])} 1}, \tag{17}$$

N is the number of state variables in Eq. (1), $\hat{x}(t)$ is the predicted response generated by sparse Bayesian regression method, and $x(t)$ is the observed response of Eq. (1). $\Delta x(t)$ represents the deviation of the predicted response and observed response at time t, which is regarded as the "distance between the two orbits at time t". Δx is the pre-selected threshold for judging the occurrence of switching, which is a reference value similar to $\xi(0)$. The basis for judging the system switching is given below:

i) When $\lambda' < 0$, it means the deviation between the predicted response and observed response does not exceed the set threshold. The system is considered to have not switched.

ii) When λ' increases sharply, from less than zero to greater than zero, we consider the system has switched.

iii) We refer to the moment corresponding to $\lambda' = 0$ as the switching moment.

3.2. The selection of optimal threshold

The threshold for judging the occurrence of switching is initialized at the beginning. If a small threshold is selected, the system will be quickly judged to have switched. At the same time, it is also possible to misjudge the occurrence of switching. Conversely, if a larger threshold is selected, the process will react slowly to the switching. Therefore, adjusting the threshold automatically for judging the occurrence of switching is important.

Algorithm 2 The pseudo code of fun_switch( )

Input: the deviation of the predicted response and observed response $\Delta x(t)$, threshold Δx and time step ΔT.
Calculate the data length of $\Delta x(t)$, denoting l.
Calculate $\lambda' = \log(\Delta x(t)/\Delta x)$.
for $j = 1{:}l$ **do**
  if $\lambda'(j) > 0$ **then**
    $t_{\text{detected}} = j \cdot \Delta T$;
    break
  end
end
Output: t_{detected} is the detected switching moment.

In this part, a series of thresholds are generated by dichotomous successive approximation algorithm. Calculate the deviation $\Delta x(t)$, given an upper threshold Δx_upper, a lower threshold Δx_lower, the real switching moment t_{switch} and precision ζ. We denote the function for calculating the detected switching moment as fun_switch( ), whose pseudo code can be seen from Algorithm 2. The switching moments t_upper and t_lower corresponding to the upper threshold and lower threshold are calculated respectively by the function fun_switch( ). After the input parameter are given, a series of thresholds are generated by successive approximation algorithm, as shown in Algorithm 3. And in the numerical simulation part of Section 4, by analyzing the effect of these thresholds on mutation detection, an optimal threshold is determined. Here, the "optimal" threshold refers to the threshold that minimizes the error between the detected switching time t_{detected} and true switching time t_{switch} while giving the input parameters of Algorithm 3 rather than optimizing a pre-specified cost function.

Algorithm 3 Dichotomous successive approximation algorithm for generating thresholds

Input: upper threshold Δx_upper, lower threshold Δx_lower, real switching moment t_{switch}, precision ζ, the deviation $\Delta x(t)$, and time step ΔT.
Calculate the switching moment: t_upper and t_lower;
Initialize count $= 0$;
while not converged and ($t_\text{lower} < t_{\text{switch}}$ && $t_\text{upper} > t_{\text{switch}}$) **do**
  count $=$ count $+ 1$;
  $\Delta x = 0.5 \times (\Delta x_\text{upper} + \Delta x_\text{lower})$;
  Calculate $t_{\text{detected}} = \text{fun_switch}(\Delta x, \Delta x(t), \Delta T)$;
  if $t_{\text{detected}} < t_{\text{switch}}$ **then**
    $\Delta x_\text{lower} = \Delta x$
  else
    $\Delta x_\text{upper} = \Delta x$
  end
  if $|\Delta x_\text{upper} - \Delta x_\text{lower}| < \zeta$ **then**
    break
  end
  thresh(count) $= \Delta x$;
  switching_t(count) $= t_{\text{detected}}$;
end
Output: thresh is a series of thresholds generated by the loop, switching_t is the switching time corresponding to the thresholds.

4. Numerical examples

In this section, two numerical examples, namely the Van der Pol system and Duffing system with parameter switching, are worked out to verify the effectiveness of the proposed switching index for mutation detection and abrupt-SBR method. The random excitation described by Gaussian white noise and periodic excitation are discussed in turn. The abrupt-SBR framework is applied to identify the renewed system equation, which is compared with the original sparse Bayesian regression method in terms of the amount of data required and the robustness against noise.

4.1. Switching Van der Pol system

4.1.1. Switching Van der Pol system without excitation

As the first example, we consider the famous Van der Pol oscillator which shows cubic nonlinearity. We change parameter values at a certain time so that the system suddenly switches its structural form. In this way, we can explore whether our method is able to accurately detect the transformation of structure and re-identify the equation of motion. The

system equations we study take the form

$$\begin{cases} \dot{x} = y - \alpha y^2, \\ \dot{y} = \mu(1-x^2)y - x + \alpha x^2, \end{cases} \tag{18}$$

Table 1. System parameters at different time scales.

Parameter	0 s–100 s	100 s–200 s	200 s–300 s
α	0.3	0	−0.4
μ	8	0.7	6

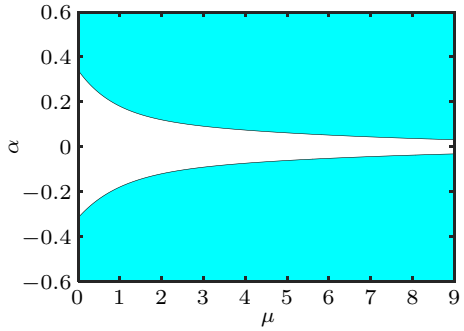


Fig. 1. The bifurcation diagram of the system for parameter α and μ. The white area indicates the system finally tends to the periodic orbit; the cyan area indicates the system finally tends to a fixed point.

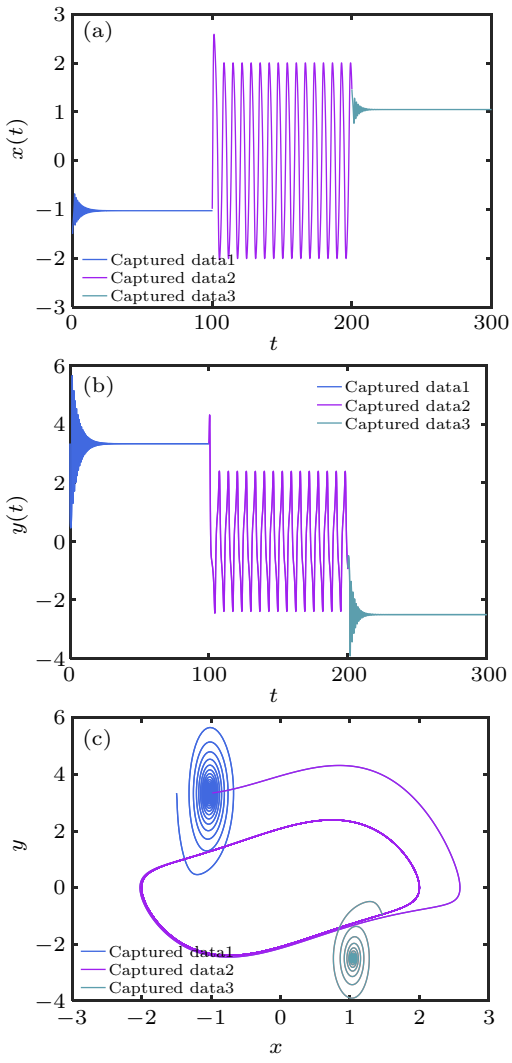


Fig. 2. Van der Pol system with $\alpha = 0.3$ and $\mu = 8$ (blue), $\alpha = 0$ and $\mu = 0.7$ (purple), $\alpha = -0.4$, and $\mu = 6$ (green). (a) The time series of x, (b) the time series of y, (c) phase space trajectory.

where α and μ are variable parameters of the system. When $\mu \in [0,9]$ and $\alpha \in [-0.6, 0.6]$, the system exhibits two dynamic behaviors. We adopt bifurcation analysis to explain the dynamic behavior of Eq. (18). Figure 1 shows the bifurcation diagram of the system for parameter α and μ, when the values of α and μ are in the white area in the figure, the system eventually tends to the periodic orbit; when it is in the cyan area, the system eventually tends to a fixed point. The changes of parameters α and μ are shown in Table 1. According to Fig. 1, we set the parameter $\alpha = 0.3$ and $\mu = 8$ for $t \in [0, 100]$ to make the system structure a focus point. At $t = 100$, the parameters are set to $\alpha = 0$ and $\mu = 0.7$ making structure become a periodic orbit. Then, the parameters change to $\alpha = -0.4$ and $\mu = 6$ at $t = 200$, which results in a focus point again. Figure 2 shows the curve of discrete data of the system under switching parameters.

I Detection of the switching occurrence

First, we identify the system equation in the first stage through sparse Bayesian regression to obtain the predicted response, and calculate the deviation $\Delta x(t)$ between the observed response and predicted response by Eq. (17). The time step in the program is 0.001. We set upper threshold $\Delta x_\text{upper} = 0.2$, lower threshold $\Delta x_\text{lower} = 0.01$, and accuracy $\zeta = 10^{-5}$. Under this set of parameter settings, a series of thresholds are generated by Algorithm 3. The iterative process of the program is shown in Fig. 3(a). Within the upper and lower limits of the threshold, the Algorithm 3 gradually approaches the optimal threshold from the left and right sides, resulting in a large shock in Fig. 3(a). We arrange the thresholds in ascending order, and redraw the change curve of switching time with threshold, as shown in Fig. 3(b). It illustrates that when the threshold is less than $\Delta x = 0.019161$, the occurrence of switching is misjudged; when the threshold is greater than $\Delta x = 0.019161$, the switching time gradually increases. Hence, under this set of parameter, the optimal threshold is 0.019161, and the corresponding switching time is only 100.83, which is very close to the real time of system switching. We take $t_{\text{switch}} = 100$ as the standard value, and the corresponding switching time t_{detected} at each threshold during the execution of the Algorithm 3 as the measurements. According to Eq. (19), the relative errors of detection at different thresholds are obtained, as shown in Figs. 3(c) and 3(d). It can be seen from Fig. 3(d) that when the threshold is less than $\Delta x = 0.019161$, the relative error keeps at a high level because the occurrence of switching is misjudged at a very early time; When the threshold is greater than $\Delta x = 0.019161$, the time delay of detection switching increases gradually, so the error also increases slowly.

$$\text{relative error} = \frac{|t_{\text{detected}} - t_{\text{switch}}|}{t_{\text{switch}}}. \tag{19}$$

Table 2. Optimal threshold under different parameters.

Δx_lower	Δx_upper	Optimal threshold
0.001	0.025	0.019164
0.001	0.25	0.019161
0.001	0.3	0.019167
0.01	0.025	0.019163
0.01	0.25	0.019163
0.01	0.3	0.019169

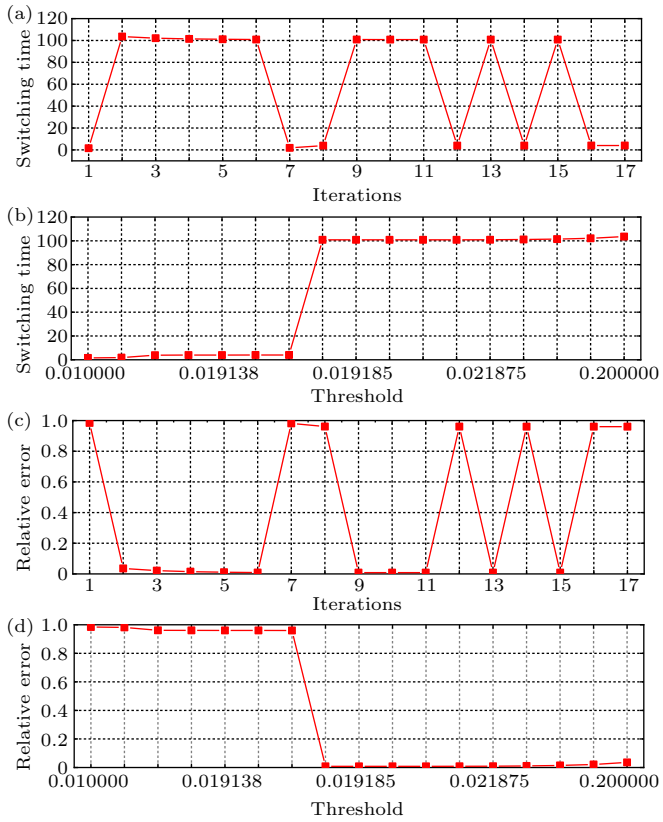


Fig. 3. (a) The change of switching time with iterations, (b) the change of switching time with threshold, (c) the change of relative error of detection with iterations, (d) the change of relative error of detection with threshold.

In order to test the robustness of the Algorithm 3, we set $\zeta = 10^{-5}$ and take six different sets of upper and lower thresholds to judge whether the difference between the obtained optimal threshold is within the range of precision ζ. Figure 4 depicts the curves of switching time and relative error with threshold under different parameters. And Table 2 lists the numerical results of the optimal threshold under the six sets of parameter settings. It can be seen that only the last significant digit of the optimal threshold is different, this is because the precision ζ of the program is 10^{-5}, which is within the allowable range of error. In addition, it can be seen from Fig. 4 that under different parameters, the position where the switching time and relative error have a sudden change are the same, which also illustrates the robustness of Algorithm 3.

It should be noted that setting a smaller precision can produce a smaller mutation detection error. As can be seen from Fig. 5, the optimal thresholds obtained under different precisions are different. The graphs in the figure depicts that with the increase of precision, the optimal threshold becomes smaller, the switching time is closer to the real switching time and relative error become smaller, indicating that there is a better detection effect.

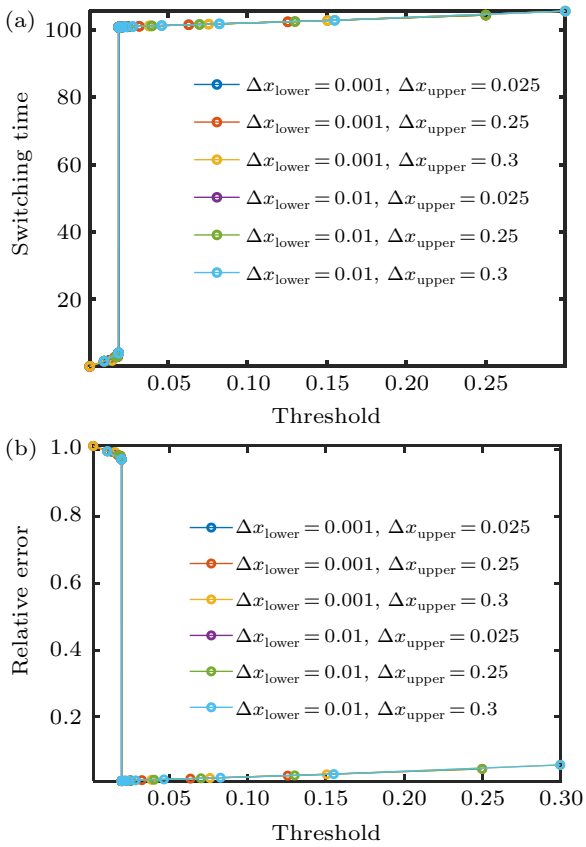


Fig. 4. Under different input parameters, the change of switching time and relative error of switching detection with threshold: (a) switching time, (b) relative error.

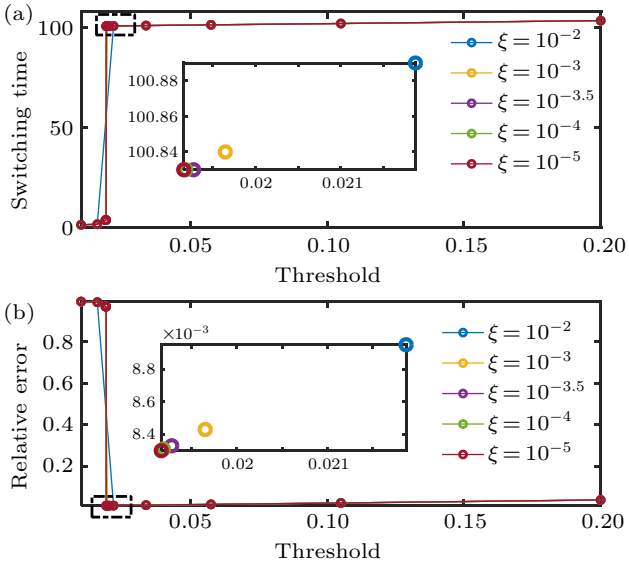


Fig. 5. Under different precisions, the change of switching time and relative error of switching detection with threshold. The graph in the figure shows the optimal threshold under different precisions. (a) Switching time, (b) relative error.

At the second switching point, we set upper threshold $\Delta x_\text{upper} = 0.001$, lower threshold $\Delta x_\text{lower} = 10^{-8}$, and precision $\zeta = 10^{-5}$, obtaining the optimal threshold $\Delta x = 7.8224 \times 10^{-6}$. The curve of switching time and relative error *versus* threshold is depicted in Figs. 6(a) and 6(b).

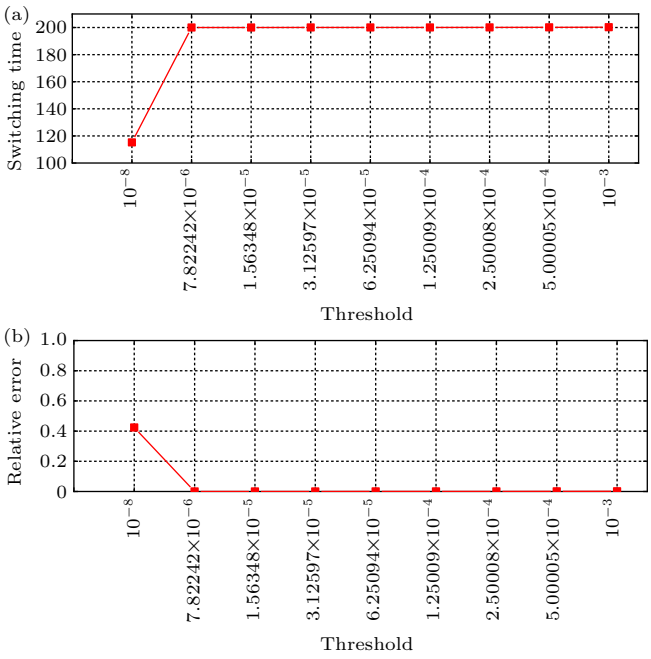


Fig. 6. The change of switching time and relative error of switching detection with threshold at the second switching point. (a) Switching time *versus* threshold, (b) relative error of switching detection *versus* threshold.

After obtaining the optimal threshold, according to Eq. (16) we can get the switching index, whose change curve is depicted in Fig. 7. At $t = 100$, the switching index increases rapidly. When the difference between the observed response and predicted response exceeds the selected threshold, the switching index begins to be greater than zero, at which point the system is detected to have switched. In the OA section, the motion equation of the next stage is predicted by the observation data. At point A, due to the re-prediction of the motion equation, the switching index rapidly decreases to a negative value.

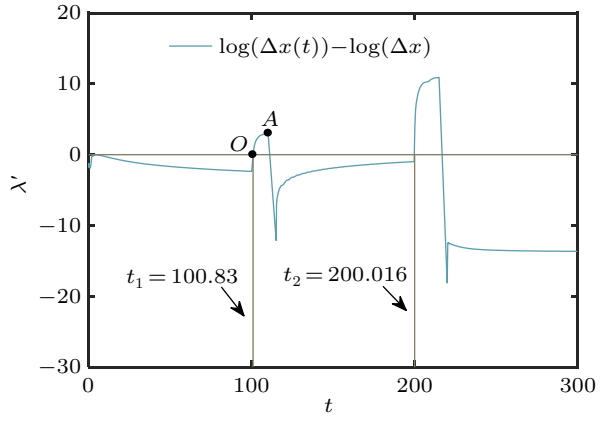


Fig. 7. The curve of the switching index with time.

II Identification of the renewed system equation

The execution of the closed-loop process of the initial identification, detection of system switching, and re-identification of the system equations enables us to track sudden changes in real time. Throughout the closed loop, abrupt changes of system parameters and structure are detected according to Eq. (16). Once the system switching is detected, we capture new discrete data and perform abrupt-SBR process to re-identify the renewed system equations:

(i) Based on the newly captured data, we perform sparse Bayesian regression on the existing sparse structure Θ_0 to get new parameter values or identify which term is deleted.

(ii) If there is an error between the predicted response and observed response, it means the term of the system increases. We perform sparse Bayesian identification process on the remaining inactive terms of Θ to find the minimum number of terms that can account for the error.

We use the abrupt-SBR framework to identify equations before and after system (18) switching. Consider the following sparse Bayesian model:

$$\begin{cases} \dot{x} = \Theta\xi_1, \\ \dot{y} = \Theta\xi_2, \end{cases} \tag{20}$$

where Θ is the basis function library, ξ_1 and ξ_2 are the weight vectors. In the identification of the first process, we build a function library where the elements are polynomials of x and y up to the third order, *i.e.*, $\{1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3\}$. After system switching, the elements in the new function library are the same as those in the equations identified in the previous stage. We set the sparse threshold $\gamma = 0.1$. According to abrupt-SBR framework, the equations of each stage identified are shown in Table 3. It is obvious that the re-identified differential equations are consistent with the real model. And the predicted trajectory can accurately and quickly track the observed trajectory, which is depicted in Fig. 8. After the first switching is detected, only 0.35 seconds of data is used to re-identify the updated system equation; After the second switching, it takes only 1.05 seconds of data to renew system equations.

The amount and noise intensity of the captured data will greatly affect the accuracy and efficiency of the identification. We investigate the effect of the amount of data on estimation error under the abrupt-SBR and standard SBR, and the results are shown in Fig. 9. $|\xi - \hat{\xi}|$ is the norm of the model coefficient difference, which is a measure of the distance between the coefficients of the predicted system and real system. Δt is the time corresponding to capturing new data, which can also be understood as the amount of data used to identify the model. Here we sample every 0.001 seconds. As shown in Fig. 9, the abrupt-SBR can identify an accurate model with a smaller amount of data than standard SBR. Meanwhile, the abrupt-SBR and standard SBR have considerable errors when

Δt is too small to provide a sufficient amount of data to identify the model. When $\Delta t = 1.6$, the models identified by these two methods have fairly small errors. However, for smaller Δt, the amount of data is too small to identify an accurate model with the standard SBR.

Table 3. The actual switching time t_{switch}, detected switching moment t_{detected}, renewed time t_{renewed}, and identified model.

t_{switch}	t_{detected}	t_{renewed}	Identified models
0	0	1.1	$\begin{cases}\dot{x}=0.999y-0.299y^2\\ \dot{y}=7.99y-7.99x^2y-1.0x+0.299x^2\end{cases}$
100	100.83	101.18	$\begin{cases}\dot{x}=0.999y\\ \dot{y}=0.70y-0.699x^2y-0.999x\end{cases}$
200	200.02	201.07	$\begin{cases}\dot{x}=0.999y+0.399y^2\\ \dot{y}=5.99y-5.99x^2y-1.0x-0.399x^2\end{cases}$

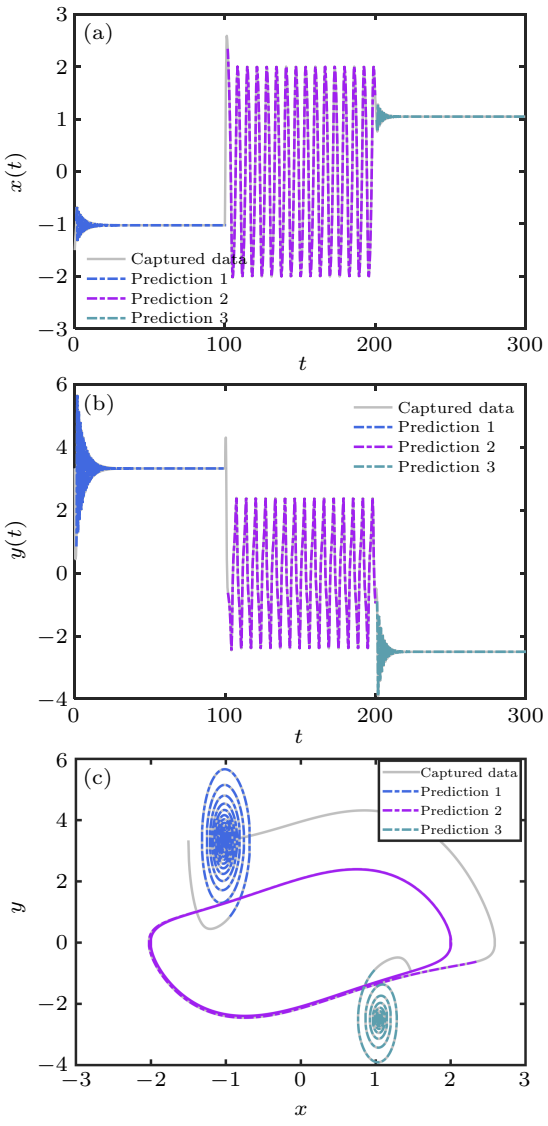


Fig. 8. Comparison of the predicted data and captured data: (a) time series of x, (b) time series of y, (c) phase space trajectory.

Then we add different levels of noise to the derivative of y to explore the effect of noise intensity on estimation error under the abrupt-SBR and SBR respectively, as depicted in Fig. 10. For the SBR algorithm, when $\Delta t = 1.5$, with the increase of noise intensity, the model error tends to increase. When the noise intensity is in the range of $[10^{-4}, 10^{-3}]$, the accurate model can be identified. However, for smaller data volumes, the SBR fails to identify the model regardless of the noise intensity. For the abrupt-SBR, it is more robust to noise than SBR. When $\Delta t = 1.5$, abrupt-SBR can handle about two orders of magnitude of noise before failing to identify an accurate model, while standard SBR only handle one order of magnitude. In addition, increasing the amount of data can reduce the error of model identification.

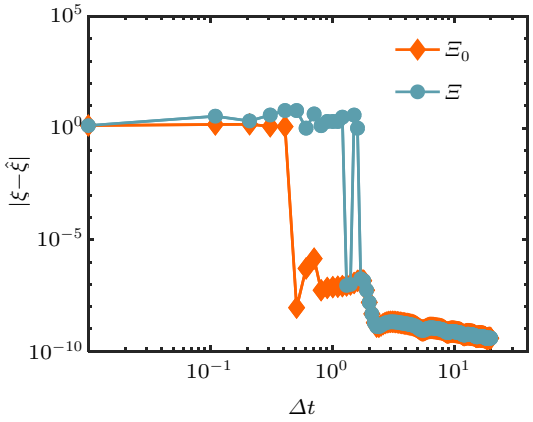


Fig. 9. Model error *versus* amount of data: the data are collected in the range of $[100.83, 100.83+\Delta t]$ after the system switching is detected the first time. The green dots represent identifying the model from scratch with SBR, and the orange diamonds represent identifying the model from the existing model with abrupt-SBR.

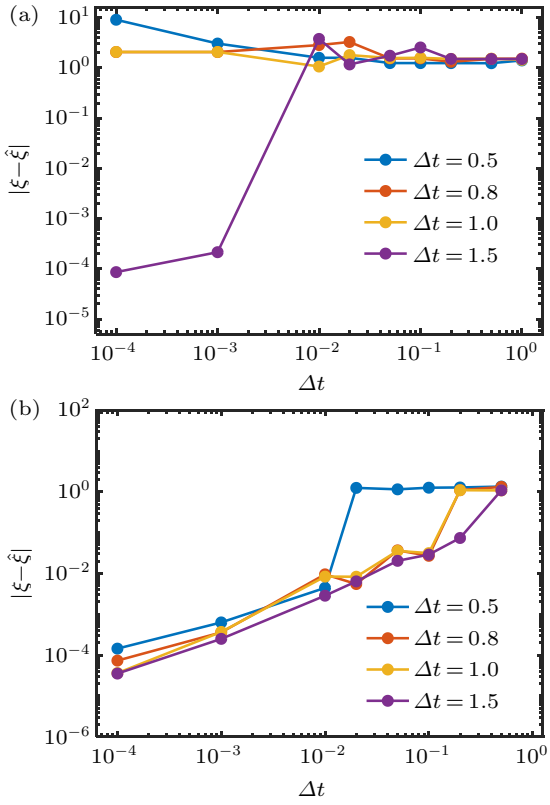


Fig. 10. Model error *versus* noise intensity at different data volumes: (a) identifying the model from scratch with SBR, (b) identifying the model from the existing model with abrupt-SBR.

Notably, we compare the abruptSBR method with the abrupt ridge regression in Ref. [36] from the perspectives of the requirement of data amount and the time to re-identify systems. In the ridge regression algorithm, we take the value of the ridge parameter as 0. First, we compare the amount of data required to identify the system under the two methods, as shown in Fig. 11. It can be seen that our method abrupt-SBR requires less data to identify an accurate model than abrupt ridge regression in Ref. [36]. The abrupt-SBR only requires $\Delta t = 0.5$, but abrupt ridge regression requires approximately $\Delta t = 2.8$. Moreover, compared with the ridge regression method, the improved abrupt ridge regression method does not greatly improve in the data volume. To illustrate the efficiency of the abrupt-SBR method in terms of the time spent on updating the system, after detecting the first switch of the system, we execute the abrupt-SBR and abrupt ridge regression algorithms on computer 20 times respectively and record the running time of the program each time. Figure 12 depicts the change curve of the program running time for 20 times under the two methods. It can be seen that the program execution time of the two methods is on an order of magnitude, and the values are very small, which shows the efficiency of abrupt-SBR in updating the system.

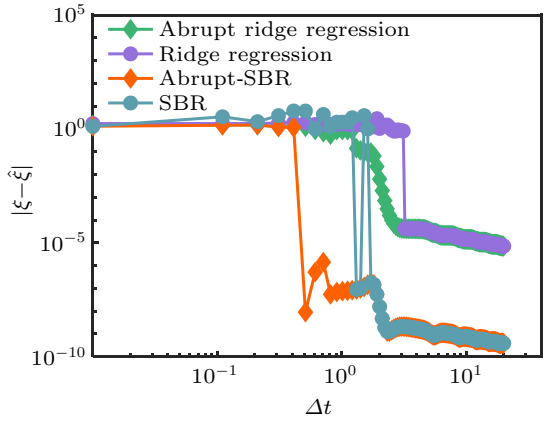


Fig. 11. Model error *versus* amount of data under different methods.

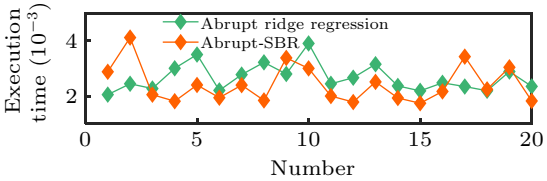


Fig. 12. The change curve of the program running time for 20 times.

4.1.2. Switching Van der Pol system with Gaussian white noise excitation

In this subsection, we consider the mutation detection and fast identification of switching Van der Pol equations excited by white Gaussian noise. The model form is as follows:

$$\begin{cases} \dot{x} = y - \alpha y^2, \\ \dot{y} = \mu(1-x^2)y - x + \alpha x^2 + \sigma\xi(t), \end{cases} \tag{21}$$

where $\xi(t)$ is Gaussian white noise, which satisfies $\langle \xi(t) \rangle = 0$, $\langle \xi(t)\xi(t+\tau) \rangle = \delta(\tau)$, σ is noise intensity. The changes of system parameters are the same as those in Table 1.

First, we take the noise intensity as $\sigma = 0.05$. The occurrence of system switching is detected by the switching index defined by Eq. (16) and the renewed system is identified by abrupt-SBR. Here, the established function library includes not only polynomials $\{1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3\}$, but also the Gaussian white noise $\xi(t)$.

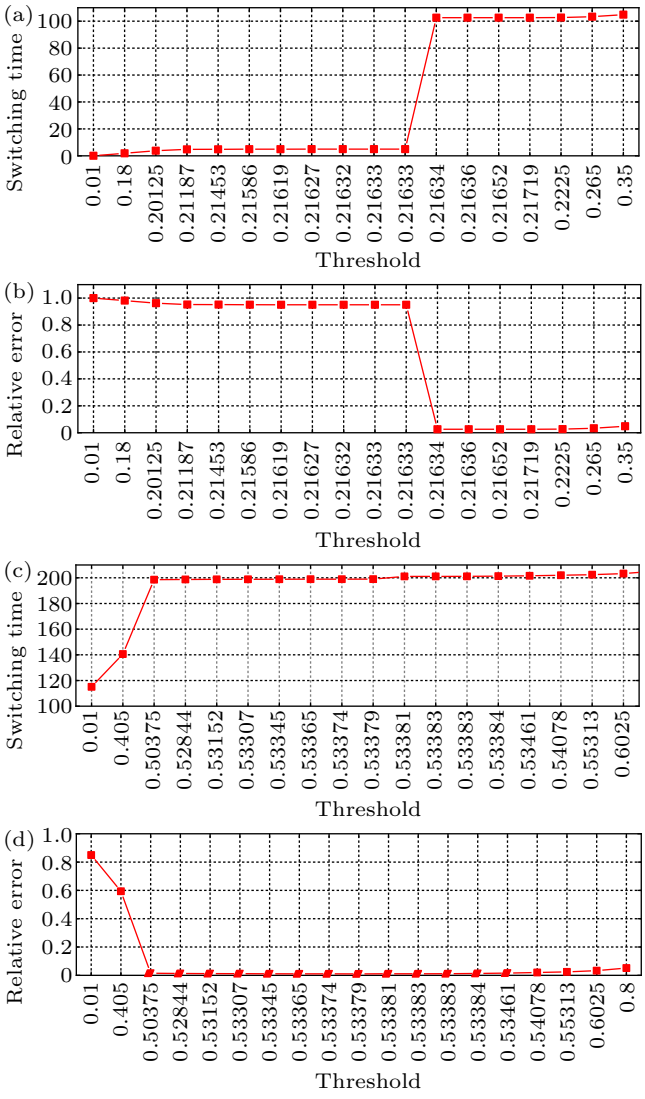


Fig. 13. The changes of switching time and relative error of switching detection with threshold: (a) switching time *versus* threshold at the first switching point, (b) relative error of switching detection *versus* threshold at the first switching point, (c) switching time *versus* threshold at the second switching point, (d) relative error of switching detection *versus* threshold at the second switching point.

We set upper threshold $\Delta x_\text{upper} = 0.35$, lower threshold $\Delta x_\text{lower} = 0.01$, and precision $\zeta = 10^{-5}$. Under this set of parameter settings, the changes of the switching time and relative error of mutation detection with threshold is shown in Fig. 13. It can be seen from Figs. 13(a) and 13(b) that the optimal threshold at the first switching point is $\Delta x = 0.21634$, where the relative error of switching detection is the smallest.

At the second switching point, when approaching the optimal threshold $\Delta x = 0.53381$ from the left, the switching time does not change significantly, which is different from the case at the first switching point. Furthermore, the optimal threshold for $\sigma = 0.05$ is larger than the optimal threshold for $\sigma = 0$ by comparing with Figs. 3 and 6. Then, the switching index is calculated by Eq. (16), and the curve with time is depicted in Fig. 14, indicating that the system is detected to have switched when $t = 102.589$ and $t = 200.13$.

The models predicted at each stage are shown in Table 4. Figure 15 depicts the comparison of the phase diagram of the predicted data and observed data. It can be seen the mutation detection method based on the switching index can successfully capture the switching of the system, and the system equation identified by abrupt-SBR is accurate, which demonstrates that our method is applicable to switching system excited by Gaussian white noise.

Table 4. The actual switching time t_{switch}, detected switching moment t_{detected}, renewed time t_{switch}, and identified model under the excitation of white Gaussian noise.

t_{actual}	t_{detected}	t_{renewed}	Identified models
0	0	1.1	$\begin{cases} \dot{x} = 0.999y - 0.299y^2 \\ \dot{y} = 7.99y - 7.99x^2y - 1.0x + 0.299x^2 + 0.0499\xi(t) \end{cases}$
100	102.59	102.99	$\begin{cases} \dot{x} = 0.999y \\ \dot{y} = 0.699y - 0.699x^2y - 0.999x + 0.0499\xi(t) \end{cases}$
200	200.13	201.63	$\begin{cases} \dot{x} = 0.999y + 0.399y^2 \\ \dot{y} = 5.99y - 5.99x^2y - 1.0x - 0.399x^2 + 0.0499\xi(t) \end{cases}$

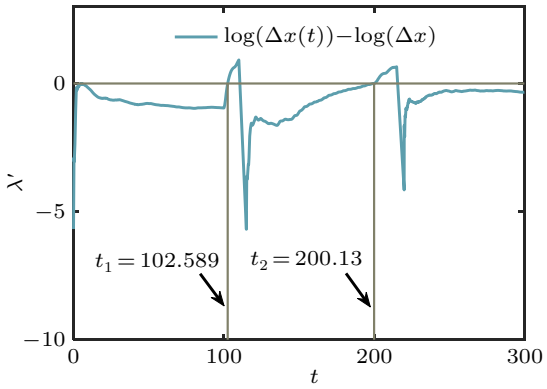


Fig. 14. The curve of the switching index with time under the excitation of white Gaussian noise.

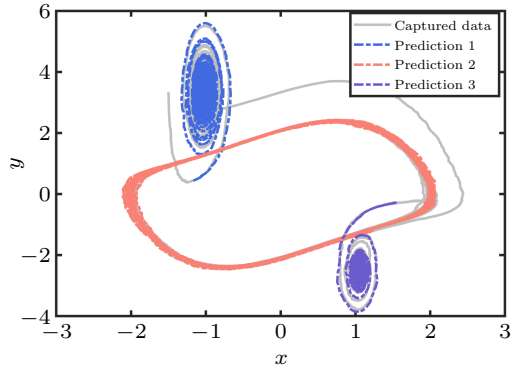


Fig. 15. Comparison of the phase lines of the predicted data and captured data under the excitation of white Gaussian noise.

Taking the first switching point as an example, in order to illustrate the robustness of the proposed mutation detection method to noise, we consider the optimal threshold under different noise intensities. Figure 16(a) depicts the change curves of switching time with threshold under different noise intensities. It indicates that as the noise intensity increases, the sudden change position of switching time gradually shifts to the right, and the ordinate value of the first point after the sudden change becomes larger, which means that the optimal threshold and switching time tend to increase monotonically, providing an inspiration for us to select an optimal threshold. That is, if the system is subject to random disturbances, the optimal threshold should be greater than the typical fluctuation range of the noise. For a larger noise intensity, a larger threshold

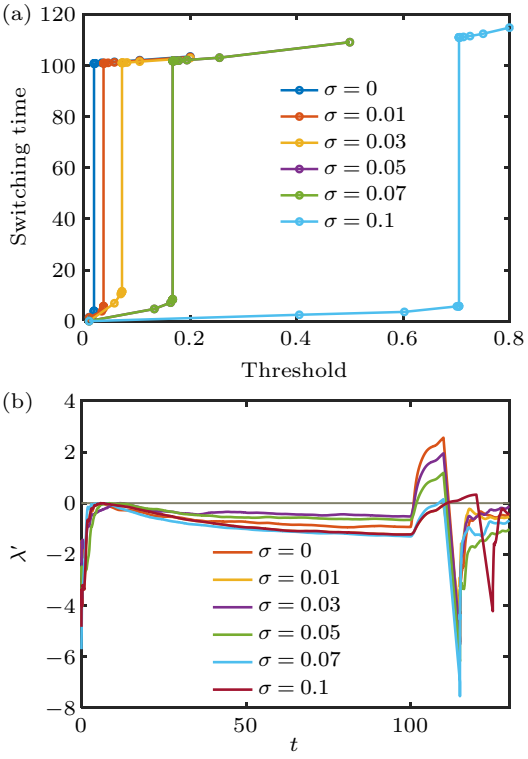


Fig. 16. (a) The curve of switching time *versus* threshold under different noise intensities, (b) the curve of switching index with time under different noise intensities.

should be selected. Figure 16(b) shows the change curves of switching index with threshold, which can be seen as the noise intensity increases, the point where the curve of the switching index first intersects with the horizontal axis gradually shifts to the right, that is, the detected switching time of the system becomes larger, which is the same as the conclusion in Fig. 16(a). Meanwhile, it also reveals the mutation detection method based on switching index has a certain degree of noise immunity.

4.2. Switching Duffing system with periodic excitation

In this subsection, we consider the famous nonlinear Duffing oscillator with periodic excitation, which shows cubic nonlinearity behavior. The control equation of the system is

$$\ddot{x}+\beta\dot{x}+\rho x+\mu x^3=F(t), \tag{22}$$

which can be transformed into a first-order system of equations by setting $\dot{x}=y$, giving

$$\begin{cases}\dot{x}=y,\\ \dot{y}=-\rho x-\beta y-\mu x^3+F(t),\end{cases} \tag{23}$$

where $F(t)=3\cos(0.5t)$. The initial condition is set as $(x_0,y_0)=(0,0.05)$. The variation of parameter values is shown in Table 5, under which the time series is depicted in Fig. 17.

Table 5. The values of system parameters at different moments.

Parameter	0 s–50 s	50 s–100 s	100 s–150 s
ρ	2	3.2	1.5
β	2.5	3.5	2
μ	0.3	2	0.3

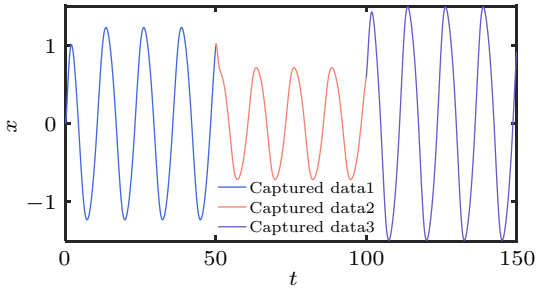


Fig. 17. Time series of x for the switching Duffing system with $\rho=2$, $\beta=2.5$, $\mu=0.3$ (blue); $\rho=3.2$, $\beta=3.5$, $\mu=2$ (orange); $\rho=1.5$, $\beta=2$, $\mu=0.3$ (purple).

In the identification of this system, we build a function library where the elements are polynomials of x, y, and F up to the third order, *i.e.*, $\{1,x,y,F,\ldots,y^3,F^3\}$, where $F=F(t)$. First, the system equations are predicted based on the observed data, and the deviation between the observed response and predicted response is calculated according to Eq. (17). At the first switching point, we set upper threshold $\Delta x_\text{upper}=0.03$, lower threshold $\Delta x_\text{lower}=10^{-5}$, and precision $\zeta=10^{-5}$. Then, Algorithm 3 is performed to generate a series of thresholds. Figures 18(a) and 18(b) depict the variation curves of the switching time and relative error with threshold, respectively. It can be seen that the optimal threshold is $\Delta x=2.46\times10^{-5}$. In this case the detected switching time is only 50.028 s, and relative error is only 0.054%. Likewise, at the second switching point, the optimal threshold is $\Delta x=3.42\times10^{-4}$. Then according to Eq. (16), we can get the switching index of the system, which is depicted in Fig. 19. At $t_1=50.028$ and $t_2=100.022$, the switching index starts to be greater than 0, which indicates the system has switched.

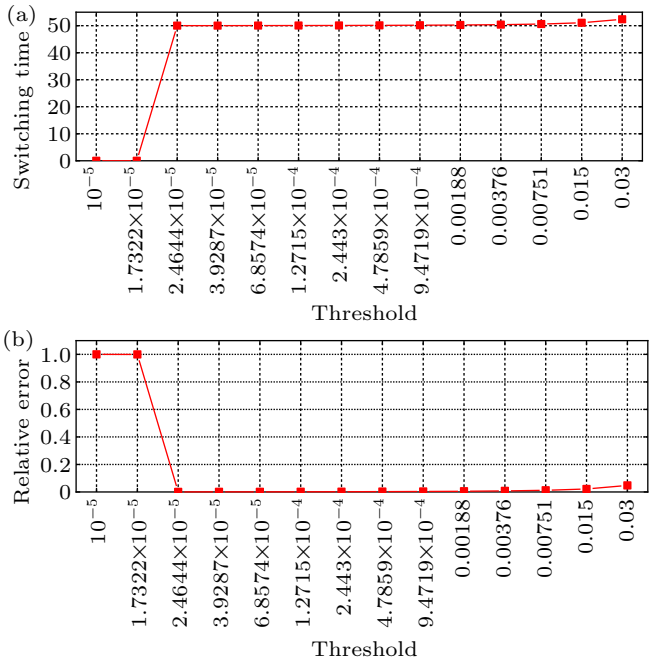


Fig. 18. (a) Switching time *versus* threshold at the first switching point. (b) Relative error of switching detection *versus* threshold at the first switching point.

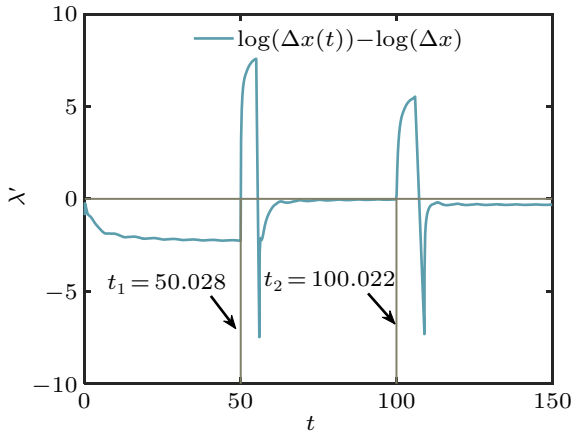


Fig. 19. The curve of the switching index with time.

When the system switching is detected, we continue to capture new discrete data and perform the abrupt-SBR process to identify the renewed system, as shown in Table 6. And the predicted trajectory can accurately and quickly track the observed trajectory, as depicted in Fig. 20.

Table 6. The actual switching time t_{switch}, detected switching moment t_{detected}, renewed time t_{renewed}, and identified model of the switching Duffing system.

t_{switch}	t_{detected}	t_{renewed}	Identified models
0	0	1	$\begin{cases}\dot{x}=0.999y\\ \dot{y}=-2x-2.499y-0.299x^3+0.999F(t)\end{cases}$
5	50.028	50.908	$\begin{cases}\dot{x}=0.999y\\ \dot{y}=-3.1998x-3.5y-2.0001x^3+0.99996F(t)\end{cases}$
100	100.022	101.822	$\begin{cases}\dot{x}=0.999y\\ \dot{y}=-1.500x-1.999y-0.2999x^3+1.000F(t)\end{cases}$

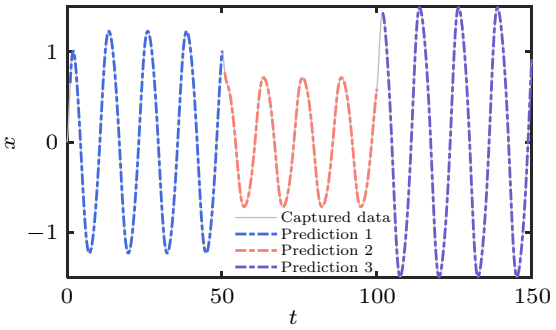


Fig. 20. Comparison of time series of x between the predicted data and captured data.

Figure 21 explores the influence of data volume on estimation error under the abrupt-SBR and SBR algorithms. Abrupt-SBR takes 0.6 seconds of data to identify an accurate model, whereas the SBR takes at least 1 second, implying that less data is required to identify the system equations based on the existing model. Then, we add different levels of noise to the derivative of y to investigate the effect of noise intensity on estimation error under different data amounts for the abrupt-SBR and standard SBR, which is depicted in Fig. 22. It is obvious that the abrupt-SBR is more robust to noise and can identify an accurate model with less data than the SBR, which is consistent with the conclusion obtained when studying the Van der Pol system.

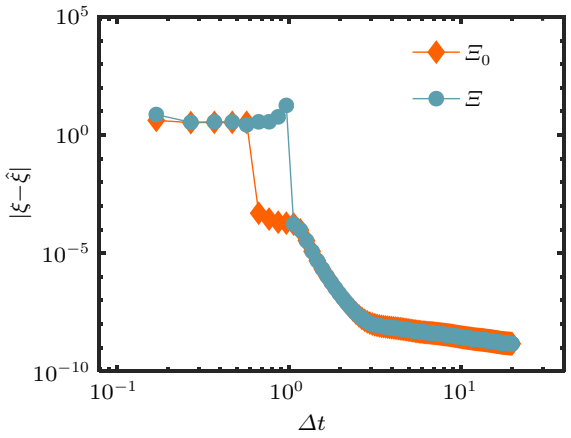


Fig. 21. Model error *versus* amount of data: the data are collected in the range of $[50.028, 50.028+\Delta t]$ after the system switching detected the first time. The green dots represent identifying the model from scratch with SBR, and the orange diamonds represent identifying the model from the existing model with abrupt-SBR.

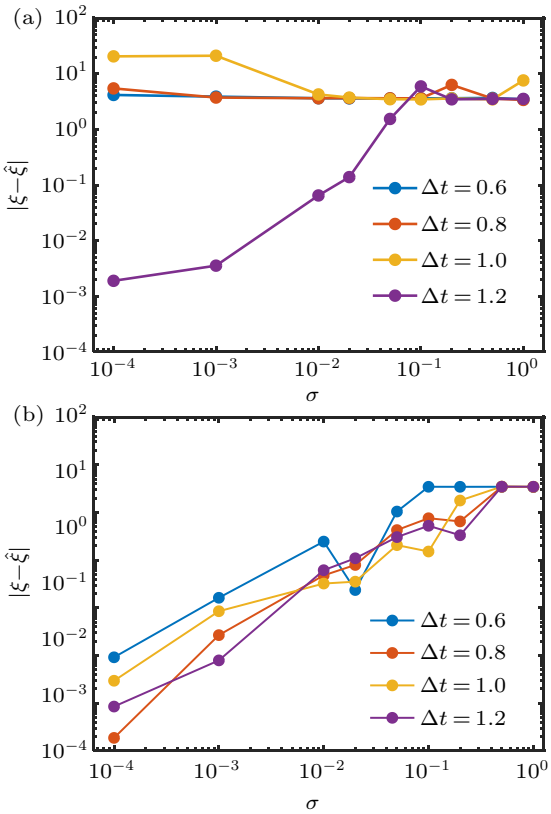


Fig. 22. Model error *versus* noise intensity at different data volumes: (a) identifying the model from scratch with SBR. (b) identifying the model from the existing model with abrupt-SBR.

5. Conclusion

To summarize, we have predicted the behavior of switching system by using the proposed new method of mutation detection and quick identification, which can successfully track the changes of the system and re-identify the system quickly. Firstly, we propose a new measure — the threshold-based switching index to detect the sudden changes of the model. We give the dichotomous successive approximation algorithm for selecting the optimal threshold, which can reduce the unnecessary data acquisition and the relative error of switching point detection. Furthermore, to demonstrate the robustness of the dichotomous successive approximation algorithm, we investigate the effect of the input parameters on the optimal threshold. The results show that the difference between the optimal thresholds is within the given precision range for dif-

ferent upper and lower threshold limits. Moreover, for different algorithm precisions, the optimal threshold generated by smaller precision is more accurate, making the mutation detection more pinpoint. Although the algorithm itself is influenced by the input parameters, the results are robust.

Secondly, after abrupt change is detected, we perform the SBR progress based on the existing model to capture the fewest changes that can account for the system, which requires less data and has better noise robustness than identifying the model from scratch, reflecting the advantages of the abrupt-SBR algorithm over the traditional SBR algorithm. Finally, two representative systems are investigated to illustrate the effectiveness and application of the proposed methods. The results show that the switching point detected is very close to the real switching point. Even if the system is subject to small random disturbance, this detection method is still effective. Meanwhile, the identified model is basically consistent with the actual model, and the predicted trajectory can track the observed trajectory quickly and accurately. Thus, the proposed method is promising to realize the mutation detection and rapid identification of switching system, which is beneficial to the accurate prediction and possible control of switching system behavior.

Furthermore, due to the sensitivity of chaotic systems to parameter values, the observed response and predicted response may diverge before the system switches. Considering the importance of mutation detection, this will be the focus of future research work. In future, we will investigate the time series of the system from the perspective of information-theoretic measurement to detect mutations.

Acknowledgment

Project supported by the National Natural Science Foundation of China (Grant No. 12072261).

References

[1] Song Y, Xu W and Niu L Z 2022 *Sci. China-Technol. Sci.* **65** 1700
[2] Zhang Y X, Jin Y F, Xu P F and Xiao S M 2020 *Nonlinear Dyn.* **99** 879
[3] Wei W, Xu W and Liu J K 2021 *Physica A* **582** 126246
[4] Yan B B, Li Y, Dai P and Liu S X 2019 *J. Aerosp. Eng.* **32** 04019058
[5] An J G, Yan M, Zhou W B, Sun X H and Yan Z 1988 *J. Aircr.* **25** 216
[6] Brunton S L and Kutz J N 2019 *Data-Driven Science and Engineering: Machine Learning* (Cambridge: Cambridge University Press) pp. 253–275
[7] Kossack C F and Kendall M G 1956 *Br. J. Stat. Psychol.* **9** 68
[8] Mann H B 1945 *Econometrica* **13** 245
[9] Yamamoto R, Iwashima T, Kazadi S N and Hoshiai M 1985 *J. Meteorol. Soc. Jpn.* **63** 1157
[10] He W P, Feng G L, Wu Q, Wan S Q and Chou J F 2008 *Nonlinear Process Geophys.* **15** 601
[11] Savit R S and Green M L 1991 *Physica D* **50** 95
[12] Pincus S 1995 *Chaos* **5** 110
[13] Abramson N, Braverman D J and Sebestyen G S 1963 *IEEE Trans. Inf. Theory* **9** 257
[14] Goodfellow I, Bengio Y and Courville A 2016 *Deep Learning* (Cambridge: The MIT Press) pp. 45–63
[15] Kalman R E 1959 *J. Fluids Eng.-Trans. ASME* **82** 35
[16] Ho B L 1965 *AT-Autom.* **14** 545
[17] Gershenfeld N 1999 *Kybernetes* **35** 597
[18] Juang J N, Pappa R S and Alvin K F 1985 *J. Guid. Control Dyn.* **8** 620
[19] Rowley C W, Mezić I, Bagheri S, Schlatter P and Henningson D S 2009 *J. Fluid Mech.* **641** 115
[20] Schmid P J 2010 *J. Fluid Mech.* **656** 5
[21] Tu J H, Rowley C W, Luchtenburg D M, Brunton S L and Kutz J N 2015 *J. Comput. Dyn.* **1** 391
[22] Kutz J N, Brunton S L, Brunton B W and Proctor J L 2016 *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems* (Philadelphia: SIAM) pp. 105–118
[23] Takeishi N, Kawahara Y and Yairi T 2017 *Proceedings of the* 31*st International Conference on Neural Information Processing Systems*, December 4–9, 2017, Los Angeles, USA, pp. 1130–1140
[24] Lusch B, Kutz J N and Brunton S L 2018 *Nat. Commun.* **9** 4950
[25] Mardt A, Pasquali L, Wu H and Noé F 2018 *Nat. Commun.* **9** 4443
[26] Wehmeyer C and Noé F 2018 *J. Chem. Phys.* **148** 241703
[27] Yeung E, Kundu S and Hodas N 2019 2019 *American Control Conference (ACC)*, July 10–12, 2019, Philadelphia, USA, pp. 4832–4839
[28] Brunton S L, Proctor J L and Kutz J N 2016 *Proc. Natl. Acad. Sci. USA* **113** 3932
[29] Mangan N M, Brunton S L, Proctor J L and Kutz J N 2016 *IEEE Trans. Mol. Biol. Multiscale Commun.* **2** 52
[30] Dam M, Brøns M, Rasmussen J J, Naulin V and Hesthaven J S 2017 *Phys. Plasmas* **24** 1
[31] Mangan N M, Kutz J N, Brunton S L and Proctor J L 2017 *Proc. R. Soc. A-Math. Phys. Eng. Sci.* **473** 20170009
[32] Boninsegna L, Nüske F and Clementi C 2018 *J. Chem. Phys.* **148** 241723
[33] Rudy S H, Brunton S L, Proctor J L and Kutz J N 2017 *Sci. Adv.* **3** e1602614
[34] Zhang Y X, Duan J Q, Jin Y F and Li Y 2021 *Nonlinear Dyn.* **106** 2829
[35] Zhang Y X, Duan J Q, Jin Y F and Li Y 2020 *Chaos* **30** 113112
[36] Markus Q, Markus A, Nathan K J and Brunton S L 2018 *Chaos* **28** 063116
[37] Hoffmann M, Froehner C and Noe F 2019 *J. Chem. Phys.* **150** 025101
[38] Huang Z L, Tian Y P, Li C J, Lin G, Wu L L, Wang Y and Jiang H Q 2020 *J. Mech. Phys. Solids* **137** 103871
[39] Li C J, Huang Z L, Wang Y and Jiang H Q 2020 *Sci. China-Technol. Sci.* **64** 148
[40] Tipping M E 2001 *J. Mach. Learn. Res.* **1** 211
[41] Bishop C M 2007 *Pattern Recognition and Machine Learning* (New York: Springer-Verlag) pp. 227–240
[42] Tipping M E and Faul A C 2003 *Ninth International Workshop on Artificial Intelligence and Statistics*, January 3–6, 2003, Florida, USA
[43] Wolf A, Swift J B, Swinney H L and Vastano J A 1985 *Physica D* **16** 285