Online frequency and amplitude tracking in structural vibrations under environment using APES spectrum postprocessing and Kalman filtering

Xuewen Yu[a], Danhui Dan*[a,b]

[a]*School of Civil Engineering, Tongji University, 1239 Siping Road, Shanghai, 200092, China*

[b]*Key Laboratory of Performance Evolution and Control for Engineering Structures of Ministry of Education, Tongji University, 1239 Siping Road, Shanghai, 200092, China*

Abstract

Identifying time-varying frequency and amplitude online in real-life structural vibrations is an essential topic of data processing in structural health monitoring. This paper proposes a novel method for this task. We assume that structural vibration signals are stationary in a short time, thus a spectral analysis method called amplitude and phase estimation is conducted to obtain the amplitude spectrum at corresponding time window, and a post-processing technique is proposed to extract the modal frequency and amplitude from the spectrum automatically. The extracted frequency and amplitude could be regarded as the average value of instantaneous frequency and the average value of instantaneous amplitude during the window, respectively. Due to the instability of measured structural vibrations and the uncertainty of spectral shapes under ambient excitation, Kalman filtering is introduced by taking the signal that reconstructed from the identified frequencies and amplitudes as the prediction to enhance the reliability and quality (signal-to-noise ratio) of the next measured signals. Numerical study is performed to inspect the performance of the proposed method. It is also employed to analyze the vibration signals of actual structures, i.e., a cable of a cable-stayed bridge, a hanger of an arch bridge and the main girder of a suspension bridge. The results show its potential to track frequency and amplitude in structural vibrations

under environmental measurements. The method is supposed to provide fundamental support for further information obtaining and high-level decision making for structural health monitoring systems.
Keywords: online frequency tracking, frequency-squeezing postprocessing, Kalman filtering, structural health monitoring

1. Introduction

In recent years, many large-scale civil structures such as bridges, buildings and dams have been equipped with structural health monitoring (SHM) systems, and massive monitoring data have been collected for condition evaluation and damage diagnosis [1, 2]. Frequency as a natural property of structures is one of the most fundamental and important indexes that should be concerned in processing monitoring data. As actual structures in service, especially for large-span bridges bearing varying live loads, exhibit nonstationary and nonlinear dynamic characteristics, many researches have been focused on the identification of time-varying frequency.

The structural vibration response is generally modeled as the superposition of multiple amplitude modulated (AM) and frequency modulated (FM) components:

$$x(t) = \sum_{k=1}^{K} x_k(t) = \sum_{k=1}^{K} a_k(t) \cos\left(\phi_k(t)\right), \tag{1}$$

where $a_k(t)$ and $\phi_k(t)$ represent the instantaneous amplitude (IA) and instantaneous phase (IP), respectively. The phase can be written as $\phi_k(t) = \int_0^t \omega_k(t)dt + \theta_k$, and then the instantaneous frequency (IF) is generally defined as the first derivative of the phase function [3]:

$$\omega_k(t) = \frac{d}{dt}\phi_k(t). \tag{2}$$

As for structure response, in Eqs. (1) and (2), the amplitude is time-varying mainly due to the randomness of ambient excitation and the energy dissipation; and the frequency is time-varying mainly due to the uncertainty (e.g., constraint uncertainty caused by tem-

perature, mass uncertainty caused by passing vehicles) and the degeneration (e.g., stiffness decrease caused by section damage, material corrosion, etc.) of structure itself. Uncertainty results in the fluctuations of frequency around a referring value (i.e., natural frequency) while degeneration results in the permanent natural frequency change that cannot recovery automatically. These fluctuations and evolution laws of frequency and amplitude can be used for fatigue analysis, damage detection [4, 5], etc. Also, they can serve as the extracted underlying features for data-driven structural diagnosis techniques [6].

There are several commonly studied ways for IF identification. One class of method is based on mode decomposition. First, the multi-component signal is decomposed into mono-components called intrinsic mode functions (IMFs) in relative literature, which can be realized by empirical mode decomposition (EMD) [7], analytical mode decomposition (AMD) [8], empirical wavelet transform (EWT) [9], variational mode decomposition (VMD) [10], etc. For more discussions and comparisons on those or other mode decomposition methods, we recommend the review works [11, 12, 13, 14] to interested readers. Second, Hilbert transform (HT) is generally employed to obtain the analytical form $z_k(t)$ of IMF $x_k(t)$ [15]:

$$z_k(t) = x_k(t) + iv_k(t), \quad v_k(t) = \mathcal{H}(x_k(t)) := \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{x_k(\tau)}{t-\tau}d\tau, \tag{3}$$

where $\mathcal{H}$ denotes the HT. The IA is obtained as the envelop of the analytical signal, $a_k(t) = \sqrt{x_k^2(t) + v_k^2(t)}$, and the phase is calculated as $\phi_k(t) = \arctan(v_k(t)/x_k(t))$. According to Eq. (2), it's able to obtain the frequency as

$$\omega_k(t) = \frac{d}{dt}\left(\arctan\left(\frac{v_k(t)}{x_k(t)}\right)\right) = \frac{x_k(t)\dot{v}_k(t) - \dot{x}_k(t)v_k(t)}{x_k^2(t) + v_k^2(t)}. \tag{4}$$

Using Eq. (4) to calculate IF avoids phase unwrapping instead of using Eq. (2) directly after obtaining the phase. For this kind of technical route, A. Baccigalupi and A. Liccardo [16], P. Ni et al. [17], Y. Xin et al. [18] respectively employed EMD, VMD and AMD to decompose the measured multi-component responses into limited number of IMFs, and the IF was identified by using HT according to Eqs.(2) and (4). Due to the end effect introduced

by HT resulting Gibbs phenomenon in IF function, P. F. Pai and R. Zhong et al. proposed the sliding-window tracking method [19] and the conjugate-pair decomposition method [20] for IF identification after the acquisition of IMFs by EMD. In addition, A. Cicone et al. [21] developed a new definition for IF by mapping the IMF $x_k(t)$ to $\theta_k(t)$ in polar coordinate to alleviate the end effect and gained better ability to characterize the local property of a signal. Besides, if the IMF is to be a regular harmonic, the energy scheme, Teager-Kaiser algorithm (TKA) [22], is also usually used, which is applicable to online computing but sensitive to noise.

The second class of method for IF identification is based on the time-frequency analysis (TFA) and consists of two steps as well. First, by TFA approaches, the original 1D signal is transformed into a 2D time-frequency (TF) plane. Then the embedded IF trajectories $\omega_k(t)$ (i.e., frequency value at each instant) in the TF plane need to be extracted for the purpose of further quantitative analysis such as natural frequency based cable force calculating [23, 24].

For the first step, the signal processing community provides a wealth of TFA tools for disclosing IF trajectories in the TF plane. Widely used include the linear TFA methods such as short time Fourier transform (STFT) and wavelet transform (WT), and quadratic TFA methods such as Winger-Vile distribution (WVD) and its variants, Cohen class distributions [25, 26]. Due to the principle of Heisenberg uncertainty, the TF representation (TFR) or spectrogram is unable to achieve high resolution in both the time and frequency directions. Many efforts have been conducted to improve the readability of TFR to present more compact and clearer IF trajectories, and this is helpful to the frequency extraction in the next step. Thus, parametric TFA methods such as chirplet transform (CT) [27] and polynomial CT (PCT) [28], which can be regarded as preprocessing techniques, were proposed by specifying the parametric form of TF atoms to match the IF's variation law of a nonstationary signal. Besides, several postprocessing techniques developed recently can greatly concentrate the energy dispersed around the IF traces in the TF plane. Reassignment method (RM) is one of the widely adopted, which relocates the coefficients of STFT along both frequency- and time-axis by replacing them with centroids of the local energy

distribution [29]. And one drawback of RM is that it doesn't support signal reconstruction. Synchrosqueezing transform (SST) proposed by Dubechies et al. [30] can be seen as a special case of RM by squeezing TF coefficients only along frequency-axis and it's allowable for signal reconstruction. Differ from RM and SST squeezing spectrogram, synchroextracting transform (SET) proposed by G. Yu [31] only extracts the coefficients of STFT most related to IF trajectories and removes the others, furtherly enhancing the concentration of TFR. Let's denote the TFR results by those TFA methods as

$$\mathrm{TFR}_{\mathrm{method}}(t,\omega), \quad \mathrm{method} \in \{\mathrm{STFT}, \mathrm{WT}, \mathrm{SST}, \mathrm{SET...}\}, \tag{5a}$$

$$i.e., \quad \mathrm{TFR}_{\mathrm{STFT}}(t,\omega) = \int_{-\infty}^{\infty} x(\tau) g^*(\tau - t) e^{-j\omega\tau} d\tau. \tag{5b}$$

For the second step, the post-task of extracting IF from the TFR is generally resolved based on the fact that coefficients at the locations of IF trajectories are greater than that at other positions around [32, 33], i.e.,

$$\hat{\omega}_k(t) = \arg\max_{\omega} \mathrm{TFR}_{\mathrm{method}}(t,\omega), \tag{6}$$

so it's also called ridge detection [34, 35]. To keep the identified IF curves continuous and smooth, many ridge detection algorithms such as dynamic path optimization [36, 37], image processing technique [38], Markov chain Monte Carlo algorithm [34], maximum gradient method [39], ant colony optimization [40], etc., have been presented recently.

Of course, these two classes of methods mentioned above are not completely independent. On the one hand, TFA methods that support signal reconstruction can also be used for mode decomposition. On the other hand, the decomposed IMFs by mode decomposition methods can also be transferred into TF plane for further IF extraction[41]. These reviewed methods are summarized in Fig. 1.

Existing studies provide powerful and plenty of ways for IF identification. However, nowadays, in the context of SHM, we are not only required to identify time-varying parame-

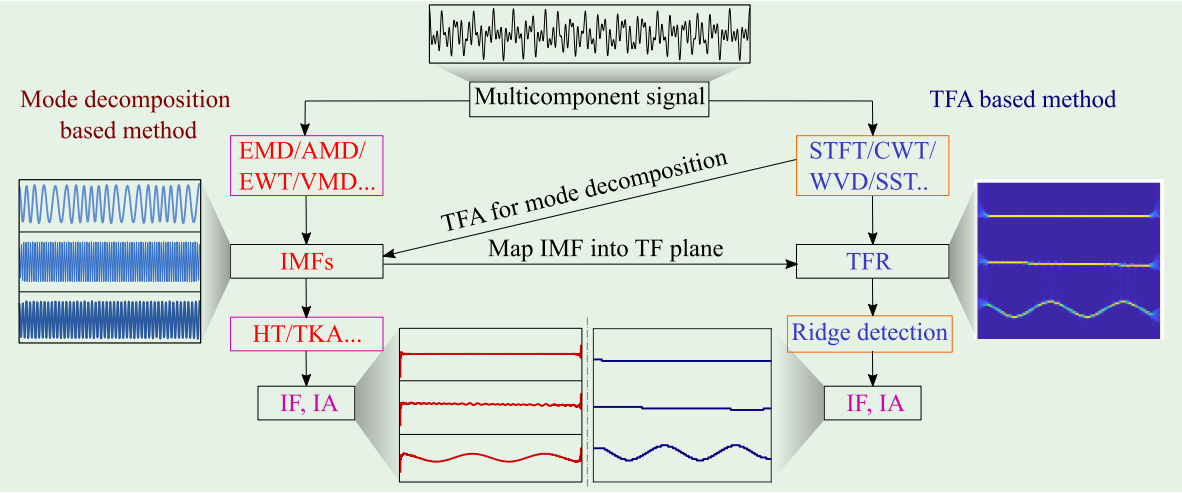

Figure 1: Review (incomplete) of existing methods of IF identification.

ters for indicating structural conditions, but also realize this task in an online and real-time manner in order to support timelier anomaly detection and decision-making[42]. This has caused the attentions of some researchers and related approaches for online modal identification and damage detection has been proposed [43, 44, 45, 46, 47, 48]. However, most existing methods as shown in Fig. 1 are realized offline with batch processing. To be adaptive to online data analysis, some modifications to them are necessary. Nevertheless, when handling with operational data, especially for the data generated by large-scale civil structures, there are some problems that need to be faced with for above methods, either realized offline or supposedly online. In online monitoring environment, there exist very weak excitation periods which means a low signal-to-noise-ratio (SNR). Obviously, since the noise is dispersed among the decomposed components except the residual term, the identified IF according to Eqs. (2) and (4) will be extremely affected. A low SNR also leads the IF trajectory in TF plane submerged in the jungle of noise, increasing the difficulty of ridge detection. And because of the randomness of excitation, the modal components participating in vibration and the number of them are varying in different periods. This becomes a further obstacle for some mode decomposition methods and ridge detection algorithms that need to specify the number of signal components in advance. Additionally, existing methods are most verified on numerical and experimental signals, lacking validity proof directly applied to actual long-time continuous signals.

This paper presents a novel approach for online frequency and amplitude identification of time-varying structural vibrations. The approach consists of two aspects, one is the automatic natural frequency and amplitude extracting from the spectrum obtained by the method of amplitude and phase estimation (APES) [49] that possesses higher resolution than FT-based methods; another is embedding this extracting algorithm for a single analytical step into a Kalman filtering (KF) framework to obtain more robust and reliable continuous tracking results. Both numerical and real-world signals prove the effectiveness of this proposed approach.

The rest of this paper is organized as follows. Section 2 defines the problem of concern and Section 3 gives a full description of the proposed solving methodology. In Section 4, synthetic signals are used to illustrate the performance of the proposed method. In Section 5, three long vibration signals that each lasts for 24 hours belong to three actual bridge structures are analyzed. The last Section 6 is the conclusion of the whole work.

2. Problem definition

The signal recorded by the sensor of an SHM system is the discretization of realistic structural vibration, suppose the sampling frequency is F_s, then the signal value at instant t is denoted as $x_t = x(t) := x(t \cdot 1/F_s)$.

This paper focuses on the online identification of frequency and amplitude and the data processing scheme is shown in Fig. 2. Unlike offline data processing: first, only past data are available; Second, assuming the start time is $t = 0$, once a sufficiently long data series (denote as $\boldsymbol{x}_0 := [x_0, x_1, \cdots, x_N]^T$ in Fig. 2(a)), we called computing frame, has been generated, the analysis of this computing frame has to start immediately; Third, the computing frame in the previous step will be updated by the new samples, called updating block, forming a new computing frame and following the analysis of it. Here, the length of computing frame is fixed as N, and denote the length of updating block as $S \in [1, N)$. Thus, as illustrated in Fig. 2(a), the online data processing is in the manner of updating the N lengthed computing frame with S newest samples on and on. Each computing frame $\boldsymbol{x}_k := [x_{kS}, x_{kS+1}, \cdots, x_{kS+N-1}]$ is

immediately analyzed to obtain the frequency and amplitude. In our paper, this is realized by using APES [49] to estimate the amplitude spectrum first and then using a postprocessing technique to extract the natural frequency and amplitude, which is introduced in Section 3.1.

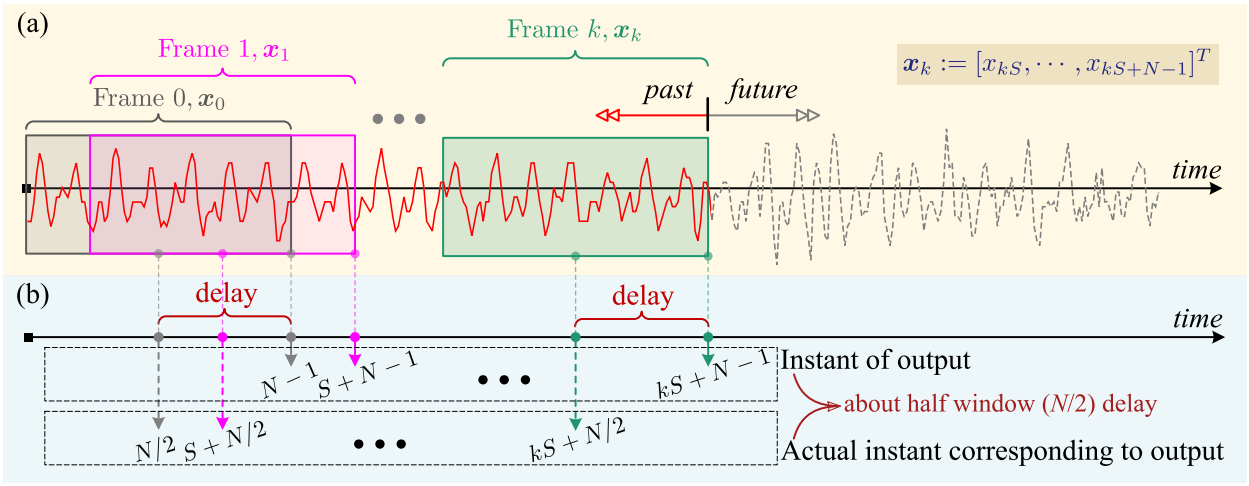

Figure 2: Online data processing scheme. (a) Fixed length computing frame and block updating model. (b) Illustration of analytical time-lag.

As shown in Fig. 2(b), the analytical result is outputted at the end of the computing frame, i.e., instants at $[N-1, S+N-1, \cdots, kS+N-1, \cdots]$. The outputting frequency is F_s/SHz. Since APES assumes the signal is stationary, the amplitude spectrum obtained by APES characterizes the average properties of the signal during the frame length, which means, the identified parameters at the end of the computing frame are actually around the center of it, i.e., instants at $[N/2, S+N/2, \cdots, kS+N/2, \cdots]$. Therefore, there exists an analysis delay about half the length of the computing frame. As for TFA methods which perform offline, the IF trajectory will also be delayed for half the window interval unless zeros are padded on both ends [50]. The delay in online analysis can be attributed to the cold start. More precisely, we define the analysis delay as the difference between the time of obtaining the parameter and the time that it actually locates, and take the time cost for data transmission and computation into accounts, then the total analysis delay $\tau_{analysis}$ can be expressed as

$$\tau_{analysis} = \tau_{method} + \tau_{computation} + \tau_{transmission} \approx \tau_{method}, \tag{7}$$

where τ_{method} represents the lag introduced by the method itself, e.g., in this paper, $\tau_{method} \approx N/2$ as mentioned above, $\tau_{transmission}$ and $\tau_{computation}$ represent the lags caused by data transmission and computation. Among them, $\tau_{method} \gg \tau_{computation} \gg \tau_{transmission}$, the analysis delay $\tau_{analysis}$ is dominated by τ_{method}.

In order to enhance the real-time performance of identification, a method that a small computing frame length N is available should be adopted. This is also one of the points that we employ APES for spectra estimate in this paper. For vibration signals of civil structures, the analysis delay is several seconds of proposed method (longer for lower frequency structures). What's more, to ensure the continuity of data processing and avoid accumulation, it has to be satisfied that

$$\tau_{computation} + \tau_{transmission} < S/F_s. \tag{8}$$

So, S is a trade-off between the output frequency F_s/S and the threshold of computation cost S/F_s.

3. Methodology

First, for every computing frame, APES is employed to estimate its spectrum and frequency squeezing postprocessing (FSP) developed in our previous work [51] is used to concentrate the spectral lines distribution. To keep the paper as self-contained as possible, we will briefly introduce the methods of APES and FSP in section 3.1. Then, a new method is proposed to automatically extract frequency and amplitude from the spectra obtained by APES and FSP (section 3.2). Last, KF is embedded in the online analysis for better parameters tracking (section 3.3).

3.1. Recaps of APES and FSP

3.1.1. APES method

APES was proposed by L. Li and P. Stoica in 1996 [49], which is a nonparametric adaptive filter-bank method with higher resolution than FT based methods [52]. Besides, it is able

to adopt shorter computing frame for exact frequency and amplitude estimation [51].

Consider a discrete one-dimensional (1D) time series $\{x(n)\}_{n=0}^{N-1}$, for a frequency ω (in Hz) of interest, it can be decomposed into a sinusoidal term with amplitude $\alpha(\omega)$ and a residual term $\varepsilon_\omega(n)$ that contains all interference caused by sinusoidal components at other frequencies as well as the random noise:

$$x(n) = \alpha(\omega)e^{j2\pi\omega n} + \varepsilon_\omega(n), \quad n = 0, 1, \cdots, N-1. \tag{9}$$

Arrange the N consecutive samples in an $M \times L$ Hankel matrix:

$$\mathbf{Y} = \begin{bmatrix} \boldsymbol{y}_0 & \boldsymbol{y}_1 & \cdots & \boldsymbol{y}_{L-1} \end{bmatrix} = \begin{bmatrix} x(0) & x(1) & \cdots & x(L-1) \\ x(1) & x(2) & \cdots & x(L) \\ \vdots & \vdots & \ddots & \vdots \\ x(M-1) & x(M) & \cdots & x(N-1) \end{bmatrix}, \tag{10}$$

where $M \leq N/2$ is a user-defined parameter, and $L = N + 1 - M$.

In the matched-filterbank interpretation of APES, the core is to design a data dependent finite-impulse-response (FIR) filter $\boldsymbol{h}(\omega) \in \mathbb{C}^{M\times 1}$ that passes the sinusoid with frequency ω undistorted and suppresses sinusoids with any other frequency as much as possible [53]:

$$\min_{\boldsymbol{h}(\omega),\alpha(\omega)} \sum_{l=0}^{L-1} |\boldsymbol{h}^H(\omega)\boldsymbol{y}_l - \alpha(\omega)e^{j2*\pi\omega l}|^2 \quad s.t. \;\; \boldsymbol{f}_M^H(\omega)\boldsymbol{h}(\omega) = 1, \tag{11}$$

where

$$\boldsymbol{f}_M(\omega) = \begin{bmatrix} 1 & e^{j2\pi\omega} & \cdots & e^{j2\pi\omega(M-1)} \end{bmatrix}^T, \tag{12}$$

and $(\cdot)^H$ and $(\cdot)^T$ denote conjugate transpose and transpose of a vector or matrix, respectively.

The closed solution to Eq. (5) is

$$\boldsymbol{h}(\omega) = \frac{\mathbf{Q}(\omega)^{-1}\boldsymbol{f}_M(\omega)}{\boldsymbol{f}_M^H(\omega)\mathbf{Q}(\omega)^{-1}\boldsymbol{f}_M(\omega)}, \tag{13}$$

$$\alpha(\omega) = \boldsymbol{h}^H(\omega)\boldsymbol{g}(\omega) = \frac{\boldsymbol{f}_M^H(\omega)\mathbf{Q}(\omega)^{-1}\boldsymbol{g}(\omega)}{\boldsymbol{f}_M^H(\omega)\mathbf{Q}(\omega)^{-1}\boldsymbol{f}_M(\omega)}, \tag{14}$$

where

$$\boldsymbol{g}(\omega) = \frac{1}{L}\sum_{l=0}^{L-1}\boldsymbol{y}_l e^{-j2\pi\omega l} = \frac{1}{L}\mathbf{Y}\boldsymbol{f}_L^*(\omega), \tag{15}$$

$$\mathbf{Q}(\omega) = \frac{1}{L}\mathbf{Y}\mathbf{Y}^H - \boldsymbol{g}(\omega)\boldsymbol{g}^H(\omega) = \mathbf{R} - \boldsymbol{g}(\omega)\boldsymbol{g}^H(\omega), \tag{16}$$

and $(\cdot)^*$ denotes conjugate of a vector or matrix.

For more details about the derivation of APES, please refer to literature [52, 53]. And we recommend the matrix implementation of it when coding as illustrated in [51] for time-saving, which is also listed in Algorithm 1 here.

Algorithm 1 APES.

1: **Input**: Data series $\{x(n)\}_{n=0}^{N-1}$, calculating frequency $\boldsymbol{\omega} = [\omega_1, \omega_2, \cdots, \omega_K]^T$, filter order M and sampling rate F_s
2: Calculate parameter $L = N + 1 - M$, matrices $\mathbf{F}_M = [\boldsymbol{f}_M(\omega_1), \boldsymbol{f}_M(\omega_2), \cdots, \boldsymbol{f}_M(\omega_K)]$ and $\mathbf{F}_L = [\boldsymbol{f}_L(\omega_1), \boldsymbol{f}_L(\omega_1), \cdots, \boldsymbol{f}_L(\omega_K)]$ by Eq. (12)
3: Construct Hankel matrix $\mathbf{Y}$ by Eq. (10) and calculate $\mathbf{R} = L^{-1}(\mathbf{Y}\mathbf{Y}^H)$
4: Calculate vectors: $\boldsymbol{a} = \mathrm{diag}(\mathbf{F}_M^H\mathbf{R}^{-1}\mathbf{F}_M)$, $\boldsymbol{b} = L^{-1}\mathrm{diag}(\mathbf{F}_M^H\mathbf{R}^{-1}\mathbf{Y}\boldsymbol{F}_L^*)$ and $\boldsymbol{c} = L^{-2}\mathrm{diag}(\mathbf{F}_L^T\mathbf{Y}^H\mathbf{R}^{-1}\mathbf{Y}\mathbf{F}_L^*)$
5: Calculate amplitude $\boldsymbol{\alpha} = \boldsymbol{b} \div (\boldsymbol{a} - \boldsymbol{a} \times \boldsymbol{c} + \boldsymbol{b} \times \boldsymbol{b}^*)$, here '$\div$' and '$\times$' denote dot products (element-wise)
6: **Output**: Frequency $\boldsymbol{\omega}$ and amplitude $\boldsymbol{\alpha}$

3.1.2. FSP method

FSP as a postprocessing technique for concentrating the spectral lines consists of three steps: (1) amplitude pretreatment, (2) frequency squeezing and (3) amplitude restoration and zero setting.

First, pretreat the APES spectrum $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \cdots, \alpha_K]^T$ by normalization and exponentiation:

$$\bar{\alpha}_i = (\alpha_i / \max(\boldsymbol{\alpha}))^m, \quad i = 1, 2, \cdots, K, \quad m > 1. \tag{17}$$

Second, squeeze uniformly distributed spectral lines by considering consecutive $2P + 1$ spectral lines $(\bar{\alpha}_{i-P}, \cdots, \bar{\alpha}_i, \cdots, \bar{\alpha}_{i+P})$ centered on frequency ω_i and replacing the value of

ω_i with the centroid of the $2P+1$ spectral lines in frequency direction:

$$\omega_i^{n+1} = \sum_{k=\max(1,i-P)}^{\min(i+P,K)} \omega_k^n \bar{\alpha}_k \; / \sum_{k=\max(1,i-P)}^{\min(i+P,K)} \bar{\alpha}_k \; , \quad i = 1, 2, \cdots, K, \quad P \geq 1. \tag{18}$$

By iteratively conducting Eq. (18), the original spectral lines will be gradually concentrated around the positions of local highest peaks. Here, the convergence criteria is set as: the frequency difference between adjacent steps should be less than a threshold δ_s, i.e., $\|\boldsymbol{\omega}^{n+1} - \boldsymbol{\omega}^n\|_2 \leq K\delta_s$. A maximum step is also set to enforce the stop of iterations.

Third, restore the amplitude that was changed in step 1 by assigning the original amplitude vector $\boldsymbol{\alpha}$ to the squeezed frequency vector $\hat{\boldsymbol{\omega}}$ in subscript order. Meanwhile, set the amplitudes at both ends of the frequency vector $\hat{\boldsymbol{\omega}}$ and that corresponding to frequencies at clusters' edges as zeros to obtain more clear result presentation. That is,

$$\hat{\alpha}_i = \begin{cases} 0, & i \in \Omega = \{1, K\} \cup \{(j, j+1) \mid \hat{\omega}_{j+1} - \hat{\omega}_j > \delta_\omega\}, \\ \alpha_i, & i \in \{1, 2, \cdots, K\} \setminus \Omega, \end{cases} \tag{19}$$

where δ_ω is a threshold used to judge the edges of frequency clusters.

The pseudo-code of FSP is also listed in Algorithm 2 here.

Algorithm 2 FSP.

1: **Input**: Frequency $\boldsymbol{\omega} = \boldsymbol{\omega}^1 = [\omega_1, \omega_2, \cdots, \omega_K]^T$ and amplitude $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \cdots, \alpha_k]^T$ by Algorithm 1, order m, step size P, thresholds δ_s and δ_ω, max iterations I, $n = 0$, $\boldsymbol{\omega}^0 = \mathbf{0}$
2: Amplitude pretreatment by Eq. (17)
3: Frequency squeezing:
4: **while** $\|\boldsymbol{\omega}^{n+1} - \boldsymbol{\omega}^n\|_2 > K\delta_s$ **and** $n < I$ **do**
5: $\quad n = n + 1$
6: $\quad$ Moving frequency by Eq. (18)
7: **end while**
8: Amplitude restoration and zero setting by Eq. (19)
9: **Output**: processed frequency $\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}^{n+1}$ and amplitude $\hat{\boldsymbol{\alpha}}$

3.2. Automatic frequency and amplitude extraction from the spectrum

For the purpose of quantitatively analyzing the changing laws of frequency and amplitude, we have to extract them from the estimated spectrum. This is usually realized by

peak analysis with a proper selection of parameters such as minimum peak separation, minimum peak height, etc. However, when handling with realistic data online, those pre-fixed parameters are not suitable for all spectrum shapes, which means not every picked peak corresponds to a natural frequency all the time. Thus, we proposed a novel approach for automatically extracting frequency and amplitude based on the original and the processed spectra.

First, it's found that the spectral lines of APES spectrum (Fig. 3(b)) are redistributed into several groups (Fig. 3(c)) after the processing of FSP, and each group corresponds to a potential location of natural frequency. So, we separate the frequency domain into P sub-intervals. In each sub-interval, the spectral lines are clustered within the maximum distance less than a threshold δ_Ω, as shown in the following equation:

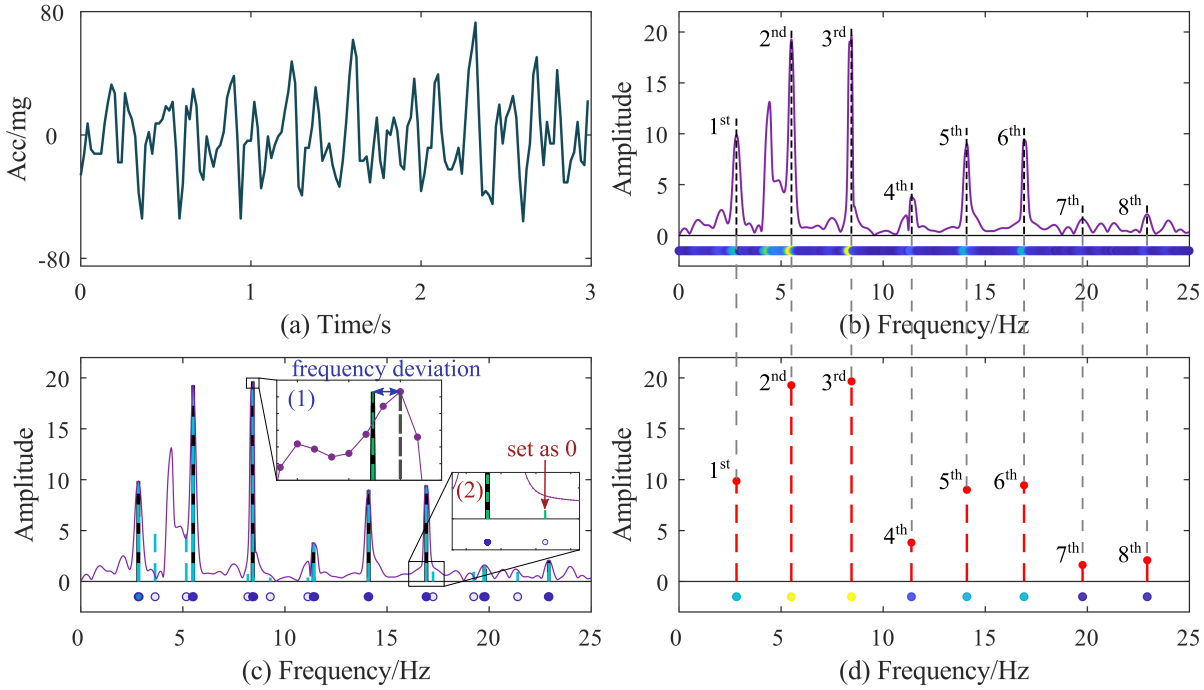

Figure 3: FSP and the extraction of frequency and amplitude. (a) Data series, which is measured from the vibration of a cable of a cable-stayed bridge. (b) APES spectrum of (a), the calculating frequency $\omega = [0, 0.01, 0.02, \cdots, 25]$Hz. (c) FSP result of (b): original spectrum (purple line); spectral lines by FSP (black line); spectral lines by FSP without zero setting of amplitude (blue dashed line). (d) Extracted natural spectral lines by Algorithm 3. The points under the spectral lines in (b)∼(d) show the distribution of frequencies.

$$\begin{cases} \Omega = \{1, 2, \cdots, K\} = \Omega_1 \cup \cdots \cup \Omega_i \cup \cdots \cup \Omega_P, \\ \Omega_i = \{j \mid |\hat{\omega}_j - \hat{\omega}_{i_0}| \leq \delta_\Omega, j \in \Omega\}, \text{where } i_0 = \begin{cases} 1, & i = 1, \\ \max(\Omega_{i-1}) + 1, & i \geq 2. \end{cases} \end{cases} \tag{20}$$

Note that the frequency $\hat{\boldsymbol{\omega}}$ still keeps monotonically increasing after FSP, i.e., $\hat{\omega}_1 \leq \hat{\omega}_2 \leq \cdots \leq \hat{\omega}_K$, and the number of sub-intervals P (i.e., the number of extracted frequency and amplitude) is automatically determined.

Second, we locate the maximum amplitude of each group Ω_i:

$$\Omega_{max} = \left\{ m_i \mid m_i = \arg\max_m(\hat{\alpha}_m), m \in \Omega_i, i = 1, 2, \cdots, P \right\}, \tag{21}$$

and index m_i of the maximum amplitude will be regarded as the position of natural frequency if it satisfies 1) $\hat{\alpha}_{m_i} \neq 0$ and 2) the distance between original frequency and processed frequency $|\hat{\omega}_{m_i} - \omega_{m_i}|$ is less than a threshold δ_d, i.e.,

$$\Omega_s = \{i \mid \hat{\alpha}_i \neq 0 \ \ \& \ \ |\hat{\omega}_i - \omega_i| < \delta_d, i \in \Omega_{max}\}. \tag{22}$$

The index set of Ω_s in Eq.(22) is still just the potential locations where natural frequencies are, because there is no guarantee that the spectral lines will not cluster at the lobes caused by noise, especially in the case of low SNR.

Finally, extracting frequencies and amplitudes from the original frequency vector $\boldsymbol{\omega}$ and amplitude vector $\boldsymbol{\alpha}$ according to the indexes set Ω_s:

$$\begin{cases} \boldsymbol{\omega}_s = \{\omega_i \mid i \in \Omega_s\}, \\ \boldsymbol{\alpha}_s = \{\alpha_i \mid i \in \Omega_s\}. \end{cases} \tag{23}$$

Here, the subscript 's' in Ω_s, $\boldsymbol{\omega}_s$ and $\boldsymbol{\alpha}_s$ means 'sparse', because most non-natural frequen-

cies and amplitudes have been removed. And we will denote the natural frequencies and amplitudes as $\boldsymbol{\omega}_n$ and $\boldsymbol{\alpha}_n$ in the following, obviously, 'n' means 'natural'. If the spectrum enjoys a well shape, each frequency extracted in Eq. (23) will correspond to a natural frequency, as shown in Fig. 3(d).

In addition, the frequency deviation in the processed spectrum $(\hat{\boldsymbol{\omega}}, \hat{\boldsymbol{\alpha}})$ by FSP as shown in 3(c) is eliminated by Eq. (23), which relocates the biased peaks to the true position in original APES spectrum $(\boldsymbol{\omega}, \boldsymbol{\alpha})$.

Actually, after the processing of FSP, the original spectral lines distributed with an equal spacing of $\Delta\omega = \omega_{i+1} - \omega_i$ will be extremely compact (e.g., at the precision of 0.001Hz or smaller) at local peaks where probably the natural frequencies are. By using this feature, the following manner of cluster separation is suggested which is more efficient than Eq. (20):

$$\check{\omega}_i = \lfloor \hat{\omega}_i \times 10^n \rfloor / 10^n, \quad i = 1, 2, \cdots, K, \tag{24a}$$

$$\acute{\boldsymbol{\omega}} = \text{unique}(\check{\boldsymbol{\omega}}) = \{\acute{\omega}_1, \acute{\omega}_2, \cdots, \acute{\omega}_P\}, \quad P \ll K, \tag{24b}$$

$$\Omega_i = \{j \mid \check{\omega}_j = \acute{\omega}_i, j \in \Omega\}, \quad i = 1, 2, \cdots, P. \tag{24c}$$

Eq. (24a) means the frequency $\hat{\omega}_i$ is truncated to the nth decimal place, and $\lfloor \cdot \rfloor$ represents round down. In Eq. (24b), operation unique$(\cdot)$ means to delete the same elements in $\check{\boldsymbol{\omega}}$, leaving only the P different elements. Eq. (24c) divides the original frequency interval into P sub-intervals.

The pseudo-code of frequency and amplitude extraction is given in Algorithm 3.

Algorithm 3 Automatic frequency and amplitude extraction.

1: **Input**: Original frequency $\boldsymbol{\omega}$ and amplitude $\boldsymbol{\alpha}$ by Algorithm 1, processed frequency $\hat{\boldsymbol{\omega}}$ amplitude $\hat{\boldsymbol{\alpha}}$ by Algorithm 2, threshold δ_Ω or order n, threshold δ_d
2: Separate the frequency domain into P sub-intervals by Eq. (20) or Eq. (24)
3: Find index of the maximum amplitude of each sub-intervals by Eq. (21) and determine if the index corresponds to a natural frequency by Eq. (22)
4: Extract frequency and amplitude through the index by Eq. (23)
5: **Output**: Extracted frequency $\boldsymbol{\omega}_s$ and amplitude $\boldsymbol{\alpha}_s$

3.3. ***Online frequency and amplitude tracking embedded with KF***

By now, using Algorithms $1 \sim 3$, it's able to automatically identify frequency and amplitude in a single step. However, when dealing with actual continuous and long-term signals of structural vibrations, it's hard to guarantee that the vibrations in every computing step are well excited by the environment. The existing signal frames with very low SNR will cause poor identification results mixed with non-natural frequency. In order to improve the naive online extracting results, i.e., each computing frame is processed separately, we apply KF to the tracking framework.

As shown in Fig. 2, denote the measured kth ($k = 0, 1, \cdots$) computing frame at time $t_k := kS + N - 1$ as $\boldsymbol{x}_k := [x_{kS}, x_{kS+1}, \cdots, x_{kS+N-1}]$, and denote the procedure of frequency and amplitude extraction from $\boldsymbol{x}_k$ using Algorithms 1$\sim$3 as FAE($\cdot$):

$$(\boldsymbol{\omega}_{s,k}, \boldsymbol{\alpha}_{s,k}) = \mathrm{FAE}(\boldsymbol{x}_k), \tag{25}$$

where, $\boldsymbol{\omega}_{s,k}$ and $\boldsymbol{\alpha}_{s,k}$ represent the estimated frequency and amplitude of the kth computing frame, respectively, and the name of operation 'FAE' means 'frequency and amplitude extraction'.

Assume that the extracted frequency and amplitude between adjacent computing frames (S samples spaced) do not change very much:

$$\boldsymbol{\omega}_{s,k+1} \cong \boldsymbol{\omega}_{s,k}, \quad \boldsymbol{\alpha}_{s,k+1} \cong \boldsymbol{\alpha}_{s,k}. \tag{26}$$

Therefore, according to Eq. (2), the vibration signal of $(k+1)$th computing frame can be written as

$$\begin{aligned} x_{k+1,n} = \mathrm{Re}\left(\sum_{i=1}^{P_{k+1}} (\alpha_{s,k+1,i}) e^{j2\pi(\omega_{s,k+1,i})n}\right) \cong \mathrm{Re}\left(\sum_{i=1}^{P_k} (\alpha_{s,k,i}) e^{j2\pi(\omega_{s,k,i})n}\right), \\ n = (k+1)S, (k+1)S+1, \cdots, (k+1)S+N-1, \end{aligned} \tag{27}$$

where $\omega_{s,k,i}$ and $\alpha_{s,k,i}$ represent the ith element of the estimated frequency vector $\boldsymbol{\omega}_{s,k}$ and

amplitude vector $\boldsymbol{\alpha}_{s,k}$ at the kth computing frame, respectively; $P_k \geq 1$ is the number of estimated parameters $\boldsymbol{\omega}_{s,k}$ and $\boldsymbol{\alpha}_{s,k}$; Re$(\cdot)$ returns the real part of a complex number. And we denote the construction operation expressed by Eq. (27) as

$$\boldsymbol{x}_{k+1} = \text{Construct}(\boldsymbol{\omega}_{s,k}, \boldsymbol{\alpha}_{s,k}). \tag{28}$$

Thus, taking the collected computing frame $\boldsymbol{x}_k$ as state vector, the prediction and measurement equations can be written as

$$\begin{cases} \boldsymbol{x}_{k+1} & = \mathcal{F}(\boldsymbol{x}_k) + \boldsymbol{w}_k, \\ \boldsymbol{z}_k & = \mathbf{H}\boldsymbol{x}_k + \boldsymbol{v}_k, \end{cases} \tag{29}$$

where, $\mathcal{F} : \boldsymbol{x}_k \to \boldsymbol{x}_{k+1}$ is realized by Eqs. (25) and (28), i.e., $\mathcal{F}(\boldsymbol{x}_k) = \text{Construct}(\text{FAE}(\boldsymbol{x}_k))$; measurement matrix $\mathbf{H}$ is selected as unit matrix $\mathbf{I}_{N\times N}$ to make a full observation; $\boldsymbol{u}_k$ is the zero-mean process white noise with covariance matrix $\text{E}[\boldsymbol{u}_k\boldsymbol{u}_k^T] = \mathbf{Q}_k$; $\boldsymbol{v}_k$ is the zero-mean measurement white noise with covariance matrix $\text{E}[\boldsymbol{v}_k\boldsymbol{v}_k^T] = \mathbf{R}_k$; and E$[\cdot]$ stands for expectation.

As $\mathcal{F}$ is a nonlinear function, generally, unscented Kalman filter (UKF) [54] or extended Kalman filter (EKF) [55] are resorted to for state estimation. For UKF, sigma points $\boldsymbol{\chi}$ are generated first[56], and then perform unscented transform on these sigma points $\boldsymbol{\chi}$ through the nonlinear function $\mathcal{F}$ yielding a set of transformed points $\boldsymbol{\mathcal{Y}} := \mathcal{F}(\boldsymbol{\chi})$ to compute the mean and covariance of the new estimate. In this procedure, function $\mathcal{F}$ composed of the operations of FAE$(\cdot)$ and Construct$(\cdot)$ has to be conducted many times, which is extremely time-consuming and is not suitable for online estimate. EKF is more efficient compared to UKF, which handles nonlinearity by locally linearizing the system at the current estimate. To linearize $\mathcal{F}(\boldsymbol{x}_k)$, we have to calculate its Jacobian matrix at $\boldsymbol{x}_k$, i.e., $\frac{\partial \mathcal{F}(\boldsymbol{x})}{\partial \boldsymbol{x}} \mid_{\boldsymbol{x}_k}$. However, this is unsolved observing the form of function $\mathcal{F}$. Considering that the state vector $\boldsymbol{x}_k \in \mathbb{R}^{N\times 1}$ in Eq. (29) represents N consecutive observations of a

vibration response (e.g. acceleration) at a measurement point of a structure. In this sense, $\boldsymbol{x}_k$ is a series of realizations of a single random variable. It's reasonable to assume that the uncertainty P of state $\boldsymbol{x}_k$ keeps the same after the mapping of $\mathcal{F}$, which equivalently means

$$\mathbf{F} := \frac{\partial \mathcal{F}(\boldsymbol{x})}{\partial \boldsymbol{x}} \mid_{\boldsymbol{x}_k} \cong \mathbf{I}_{N\times N}, \tag{30}$$

and the covariance matrices of process white noise and measurement white noise are reduced to a variance: $\mathbf{Q}_k \to Q_k := \mathrm{E}[\boldsymbol{u_k}^T \boldsymbol{u_k}]$, $\mathbf{R}_k \to R_k := \mathrm{E}[\boldsymbol{v_k}^T \boldsymbol{v_k}]$.

Therefore, a standard liner Kalman filter is formalized as follows:

$$(\boldsymbol{\omega}_{s,k}, \boldsymbol{\alpha}_{s,k}) = \mathrm{FAE}(\hat{\boldsymbol{x}}_{k|k}), \tag{31a}$$

$$\hat{\boldsymbol{x}}_{k+1|k} = \mathcal{F}(\hat{\boldsymbol{x}}_{k|k}) = \mathrm{Construct}(\boldsymbol{\omega}_{s,k}, \boldsymbol{\alpha}_{s,k}), \tag{31b}$$

$$\tilde{\boldsymbol{y}}_{k+1} = \boldsymbol{z}_{k+1} - \hat{\boldsymbol{x}}_{k+1|k}, \tag{31c}$$

$$e_{k+1} = \|\hat{\boldsymbol{x}}_{k+1|k}\|_2^2 / (\|\hat{\boldsymbol{x}}_{k+1|k}\|_2^2 + \|\boldsymbol{z}_{k+1}\|_2^2), \tag{31d}$$

$$d_{k+1} = 1 - \hat{\boldsymbol{x}}_{k+1|k}^T \boldsymbol{z}_{k+1} / \sqrt{(\hat{\boldsymbol{x}}_{k+1|k}^T \hat{\boldsymbol{x}}_{k+1|k})(\boldsymbol{z}_{k+1}^T \boldsymbol{z}_{k+1})}, \tag{31e}$$

$$Q_{k+1} = (\rho(1 - e_{k+1}) + (1-\rho) d_{k+1}) \sigma_{\tilde{\boldsymbol{y}}_{k+1}}, \tag{31f}$$

$$R_{k+1} = (\rho e_{k+1} + (1-\rho)(1 - d_{k+1})) \sigma_{\tilde{\boldsymbol{y}}_{k+1}}, \tag{31g}$$

$$P_{k+1|k} = P_{k|k} + Q_{k+1}, \tag{31h}$$

$$K_{k+1} = P_{k+1|k} / (P_{k+1|k} + R_{k+1}), \tag{31i}$$

$$\hat{\boldsymbol{x}}_{k+1|k+1} = \hat{\boldsymbol{x}}_{k+1|k} + K_{k+1} \tilde{\boldsymbol{y}}_{k+1}, \tag{31j}$$

$$P_{k+1|k+1} = (1 - K_{k+1}) P_{k+1|k}. \tag{31k}$$

The process function $\mathcal{F}$ is separated into two parts of Eqs. (31a) and (31b) to explicitly express the extraction of frequencies and amplitudes. Eq. (31c) computes the residual between prediction $\hat{\boldsymbol{x}}_{k+1|k}$ and measurement $\boldsymbol{z}_{k+1}$. Eqs. (31d)$\sim$(31g) aim to computing the variance Q_{k+1} of process noise and the covariance R_{k+1} of measurement noise adaptively based on the following considerations:

(1) If the measurement $\boldsymbol{z}_{k+1}$ enjoys a high energy (defined as $\|\boldsymbol{z}_{k+1}\|_2^2$), which means the structure is well excitated and the measured signal has a high SNR, thus we should be inclined to trust the measurement to capture changes in the signal's spectral structure. Otherwise, we should be inclined to the prediction $\hat{\boldsymbol{x}}_{k+1|k}$ to keep the continuity of identification results. This defines the energy metric e as Eq. (31d).

(2) If the prediction $\hat{\boldsymbol{x}}_{k+1|k}$ is similar to the measurement $\boldsymbol{z}_{k+1}$, we are inclined to the prediction, because it is reconstructed by extracted frequencies and amplitudes, meaning the noise term has been greatly filtered. Otherwise, we should be inclined to the measurement $\boldsymbol{z}_{k+1}$ to ensure the sensitivity to change perception. This defines the distance metric d as Eq. (31e).

Therefore, Q_{k+1} (process noise's variance) and R_{k+1} (measurement noise's variance) are designed relating to above energy metric e and distance metric d as shown in Eq. (31f) and Eq. (31g), where $0 \leq \rho \leq 1$ is a weight between the energy metric and distance metric and $\sigma_{\hat{\boldsymbol{y}}}$ is the variance of residual term $\hat{\boldsymbol{y}}$. Eqs. (31h) and (31i) are to obtain the Kalman gain according to the process uncertainty and the measurement uncertainty. Eq. (31j) gives the state estimate and Eq. (31k) gives the estimate of process uncertainty according to Riccati equation [57]. The form of Eq.(31) can be analogous to the standard univariate Kalman filtering.

As shown in Eq. (31), we take the structural vibration response at a certain measured point in time-domain as the state variable, and the actually concerned natural frequency and amplitude in frequency-domain are exported from the state estimate. The frequency $\boldsymbol{\omega}_{s,k}$ and corresponding amplitude $\boldsymbol{\alpha}_{s,k}$ identified by Eq. (31a) or Eq. (25) are arranged in order of frequency from the smallest to the largest, i.e., $\omega_{s,k,1} < \omega_{s,k,2} < \cdots < \omega_{s,k,P_k}$. Due to the randomness of environment, there is no guarantee that all modes are excited efficiently always, and we couldn't make sure that the identified frequency $\boldsymbol{\omega}_{s,k}$ and amplitude $\boldsymbol{\alpha}_{s,k}$ at different steps are in alignment. Alignment here has two meanings: (1) the number P_k of identified frequency $\boldsymbol{\omega}_{s,k}$ in each step keeps the same: $P_k = P_{k+1}$; (2) at the same indexed

position i, the element $\omega_{s,k,i}$ of vector $\boldsymbol{\omega}_{s,k}$ in each step is of the same order. If frequencies in different steps are not aligned all the time, it is hard to conduct further quantitative analysis but for result visualization.

Fig. 4 demonstrates the frequency alignment schema designed in this paper. First, the reference frequency $\boldsymbol{\omega}_r$ and the allowable variation range $\boldsymbol{\delta}_a$ have to be appointed in advance. For each reference position i, select the candidates from extracted frequency $\boldsymbol{\omega}_{s,k+1}$, which are in the supposed interval $(\omega_{r,i} - \delta_{a,i}, \omega_{r,i} + \delta_{a,i})$:

$$\Omega_i = \{j \mid |\omega_{s,k+1,j} - \omega_{r,i}| < \delta_{a,i}, j \in \{1, 2, \cdots, P_{k+1}\}\}, \quad i = 1, 2, \cdots, P_r. \tag{32}$$

In the above equation, P_{k+1} means the number of extracted frequency $\boldsymbol{\omega}_{s,k+1}$ at the $k+1$th computing frame, P_r means the number of reference frequency $\boldsymbol{\omega}_r$, Ω_i is the index set of candidates correspond to $\omega_{r,i}$.

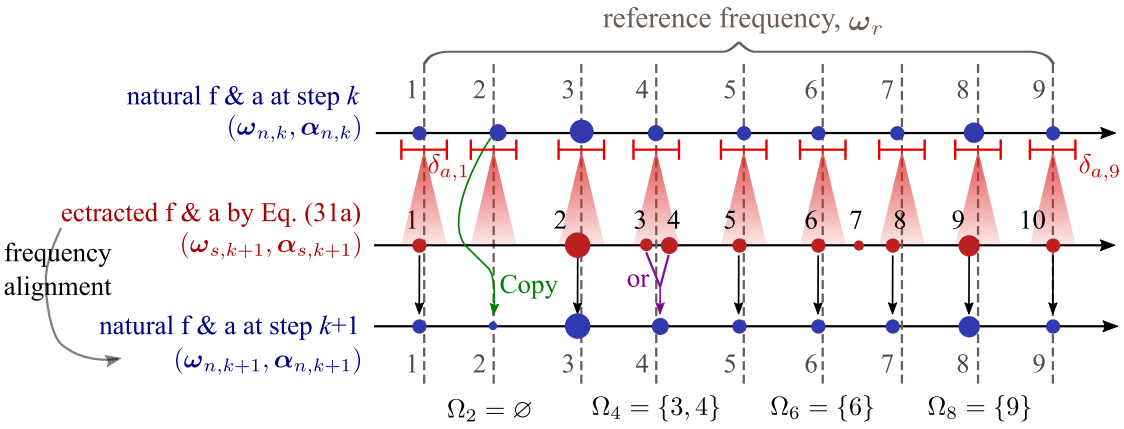

Figure 4: Illustration of frequency alignment in Eqs. 32 and 33. 'f & a' in the figure means 'frequency and amplitude'.

Second, determine true natural frequencies and amplitudes from the candidates: if there is no candidate for $\omega_{r,i}$, i.e., $\Omega_i = \varnothing$, the frequency will be copied from the last step, and the amplitude will be assigned as half of that in the last step; If the candidates are unique for $\omega_{r,i}$, they will be directly set as the natural frequency and amplitude; If there are multiple candidates for $\omega_{r,i}$, first select the candidate with maximum amplitude, $\Omega_{i,max} = \arg\max_j\{\alpha_{s,k+1,j}, j \in \Omega_i\}$, and the candidate that is nearest to the frequency $\omega_{n,k,i}$ in the last step, $\Omega_{i,nearest} = \arg\min_j\{|\omega_{s,k+1,j} - \omega_{n,k,i}|, j \in \Omega_i\}$, then if the maximum amplitude

$\alpha_{s,k+1,\Omega_{i,max}}$ is greater than twice the amplitude $\alpha_{n,k,i}$ in the last step, the frequency and amplitude will be set according to the maximum candidate $\Omega_{i,max}$, otherwise, they are set according to the nearest candidate $\Omega_{i,nearst}$. Those are expressed in the following equation:

$$\omega_{n,k+1,i} = \begin{cases} \omega_{n,k,i} & \Omega_i = \varnothing, \\ \omega_{s,k+1,\Omega_i} & \mathrm{card}(\Omega_i) = 1, \\ \begin{cases} \omega_{s,k+1,\Omega_{i,max}} & \alpha_{s,k+1,\Omega_{i,max}} \geq 2\alpha_{n,k,i} \\ \omega_{s,k+1,\Omega_{i,nearest}} & \text{otherwise} \end{cases} & \mathrm{card}(\Omega_i) \geq 2, \end{cases} \quad i = 1, 2, \cdots, P_r, \tag{33a}$$

$$\alpha_{n,k+1,i} = \begin{cases} 0.5\alpha_{n,k,i} & \Omega_i = \varnothing, \\ \alpha_{s,k+1,\Omega_i} & \mathrm{card}(\Omega_i) = 1, \\ \begin{cases} \alpha_{s,k+1,\Omega_{i,max}} & \alpha_{s,k+1,\Omega_{i,max}} \geq 2\alpha_{n,k,i} \\ \alpha_{s,k+1,\Omega_{i,nearest}} & \text{otherwise} \end{cases} & \mathrm{card}(\Omega_i) \geq 2, \end{cases} \quad i = 1, 2, \cdots, P_r, \tag{33b}$$

where, $\mathrm{card}(\cdot)$ returns the number of elements in the set.

For the situation where the time-varying frequency fluctuates at a certain level, the reference value of frequency can be determined by the statistical average based on the analysis results of historical data, and it can be updated according to the latest analysis results to capture the slow change. For the situation where the frequency shows obvious trend variation, we can set the extracted results in the last step as the reference value of frequency, i.e., $\boldsymbol{\omega}_r \leftarrow \boldsymbol{\omega}_{n,k}$.

Algorithm 4 gives the pseudo-code of frequency and amplitude tracking embedded with Kalman filtering.

Algorithm 4 Frequency and amplitude tracking using APES postprocessing and KL.

Input: Initial frequency $\boldsymbol{\omega}_{n,0}$ and amplitude $\boldsymbol{\alpha}_{n,0}$, updating block length S, weight ρ, reference frequency $\boldsymbol{\omega}_r$ and allowable frequency variation range δ_a
for $k = 0, 1, 2, \cdots$ **do**
  Predict the state $\hat{\boldsymbol{x}}_{k+1|k}$ by Eq. (31b)
  Obtain the measurement $\boldsymbol{z}_{k+1} = [x_{kS}, x_{kS+1}, \cdots, x_{N+kS-1}]$
  Calculate the residual signal $\tilde{\boldsymbol{y}}_{k+1}$ by Eq. (31c)
  Calculate the variance of process noise Q_{k+1} and the variance of measurement noise R_{k+1} by Eqs. (31d)$\sim$(31g)
  Calculate the gain K_{k+1} by Eqs. (31h)$\sim$(31i)
  Update the state $\hat{\boldsymbol{x}}_{k+1|k+1}$ by Eq. (31j) and the process uncertainty $P_{k+1|k+1}$ by Eq. (31k)
  Extracted the frequency $\boldsymbol{\omega}_{s,k+1}$ and amplitude $\boldsymbol{\alpha}_{s,k+1}$ by Eq. (31a), i.e., Algorithms 1$\sim$3
  Frequency alignment by Eqs. (32) and (33) to obtain the natural frequency $\boldsymbol{\omega}_{n,k+1}$ and amplitude $\boldsymbol{\alpha}_{n,k+1}$.
end for
Output: Time-varying natural frequency $[\boldsymbol{\omega}_{n,0}, \boldsymbol{\omega}_{n,1}, \cdots]$ and amplitude $[\boldsymbol{\alpha}_{n,0}, \boldsymbol{\alpha}_{n,1}, \cdots]$

*3.4. **Section summary***

3.4.1. Parameter settings in the algorithms

The proposed approach involves Algorithms 1$\sim$4. A specific description of the parameter settings in those Algorithms is provided in Appendix A.

3.4.2. Usages

The proposed method consists of Algorithms 1$\sim$4 could be employed for different purposes in different ways, in addition to the online frequency and amplitude tracking, which is the most focus of this paper, as illustrated in Fig. 5.

1. Using Algorithms 1 and 2 (APES+FSP) for TFR obtaining. This combination is equivalent to SST or SET in terms of result exhibition, as shown in Fig. 5(a). And this is exactly the work we done in [51]. Because the spectral lines of a TFR slice are concentrated at natural frequencies by frequency squeezing, so we called the result squeezed TFR.
2. Using Algorithms 1$\sim$3 (defined as FAE in Eq. (25)) for sparse TFR obtaining. In this combination, the potential spectral lines corresponding to natural frequencies are further extracted on the base of squeezed TFR. That is, the number of spectral lines

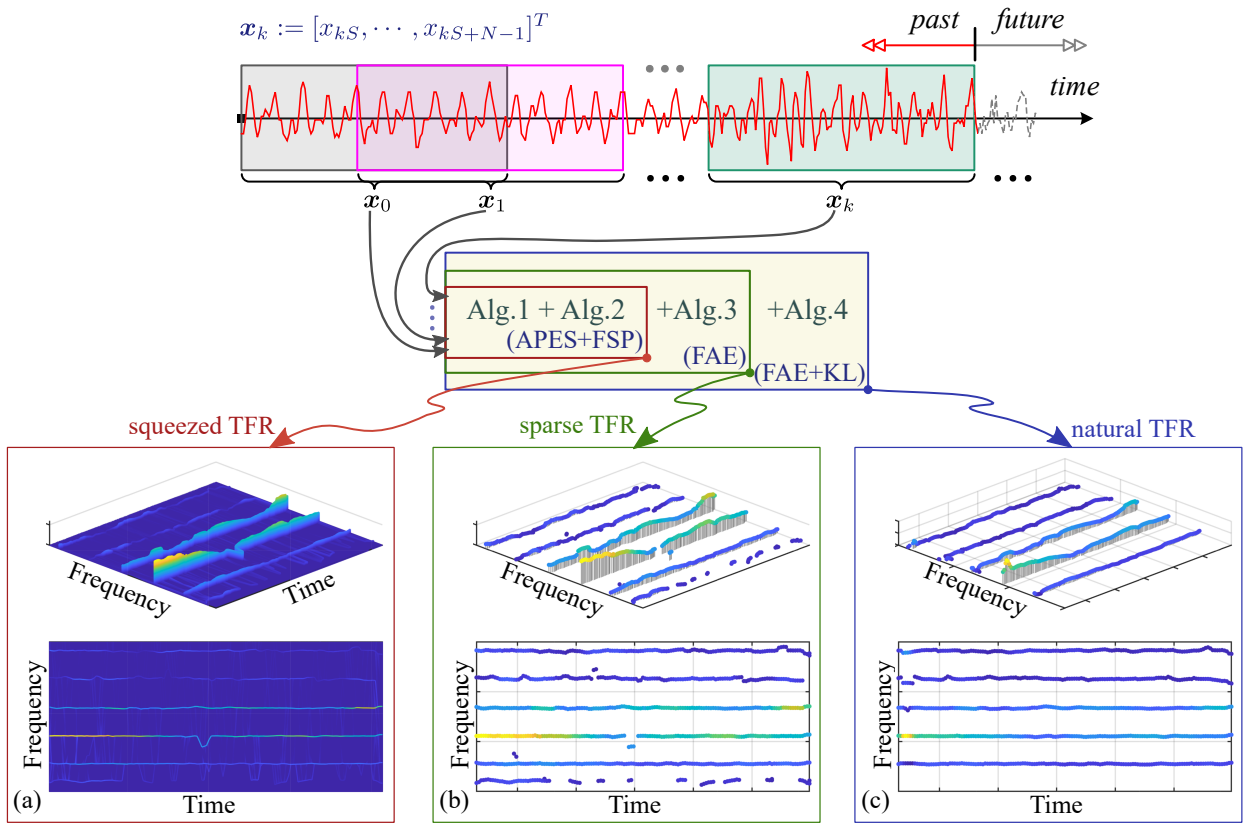

Figure 5: Usages of the proposed Algorithms. (a) Algorithms 1 and 2 for squeezed TFR. (b) Algorithms 1∼3 for sparse TFR. (c) Algorithms 1∼4 for natural frequency and amplitude tracking.

of each slice is reduced from K to P_k, and $P_k \ll K$, so we called the result here sparse TFR, as shown in Fig. 5(b).

3. Using Algorithms 1∼4 (FAE+KL) for online frequency and amplitude tracking. In which, the Kalman filtering enhances the reliability of sparse TFR and the adopted frequency alignment removes all non-natural spectral lines, as shown in Fig. 5(c).

We illustrate the usages of 1 and 2 here is that sometimes the whole TFR plotting gives us more comprehensive and convincing results than the only surviving frequency trajectories. However, the latter supports further quantitative analysis.

3.4.3. Characteristics

The main characteristics of the proposed method are described below.

1. The method is applicable to online frequency and amplitude tracking with a small analysis lag and high resolution due to the superiority of APES. And it's data-dependent

in two aspects: one is the data-dependent 'basis' of APES spectra analysis and the other is the adaptive noise variance setting in Kalman filtering.

2. The framework is easily altered and to derive other specific approaches, such as replacing APES with other spectral analysis methods, adopting other frequency alignment strategies, establishing the state equation in Kalman filtering with other forms.
3. The signal reconstruction using the identified frequency and amplitude can be regarded as the filtering of measured signal.
4. The frequency alignment requests that the different modes are well separated, this is true for most civil structures.

4. Numerical verification

Because the true values of frequency and amplitude of realistic signals are unknown, before diving to them, we investigate a numerical example to explicitly illustrate the performance of the proposed method on accuracy and noise robustness.

The considered synthetic signal with a sampling rate of 50Hz consists of three components and an additive Gaussian white noise term:

$$\begin{aligned} &x(t) = x_1(t) + x_2(t) + x_3(t) + n(t), \quad 0 \le t \le 40, \\ &x_1(t) = 3\cos(2\pi t + 2\sin(0.25\pi t)), \quad x_2(t) = 4e^{-0.05t}\cos(5\pi t - 0.005\pi t^2), \\ &x_3(t) = \begin{cases} 3.0\cos(8\pi t), & 0 \le t < 10, 20 \le t < 30, \\ 1.5\cos(8\pi t), & 10 \le t < 20, 30 \le t < 40, \end{cases} \quad n(t) \sim \mathcal{N}(0, \sigma_n^2), \end{aligned} \tag{34}$$

where σ_n denotes the standard deviation of the white nose. In this section, a low noise level ($\sigma_n = 0.5$) case in which SNR=21.52dB and a high noise level ($\sigma_n = 2.0$) case in which SNR=-2.57dB are analyzed. And the SNR is calculated as

$$\mathrm{SNR} = 10\log_{10}\frac{\|\mathrm{signal}\|_2^2}{\|\mathrm{noise}\|_2^2}. \tag{35}$$

Fig. 6 plots the components, noise term, synthetic signal and IF laws defined in Eq. (34). The IA functions can be seen from the envelops of components in Figs. 6(a)∼(c).

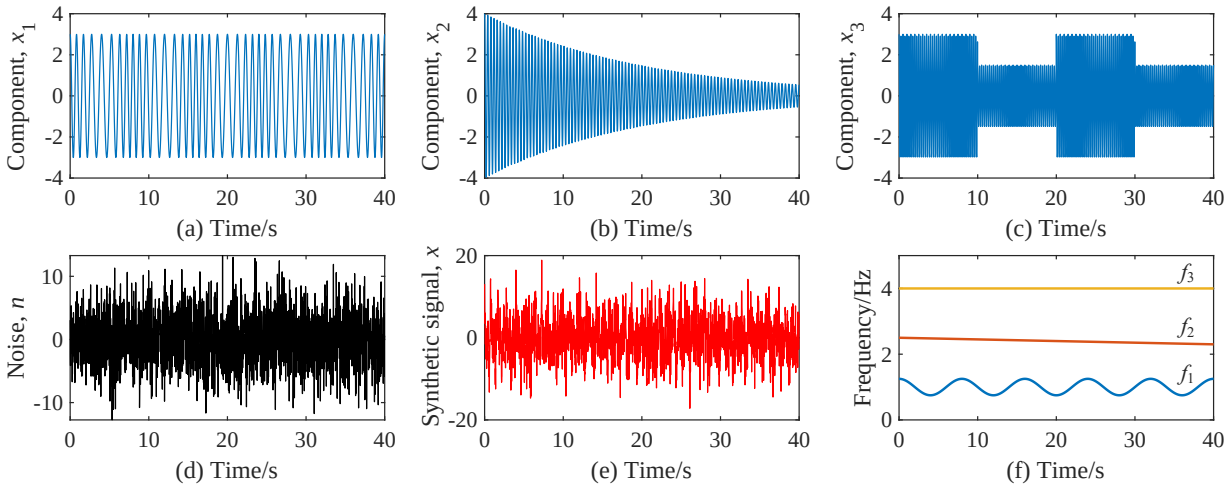

Figure 6: Numerical signal in Eq. (34). (a)∼(c) Components 1∼3. (d) Noise term (showed $\sigma_n = 2.0$). (e) Synthetic signal. (d) IF law.

We compare the proposed method with SET to validate its effectiveness. The frequency and amplitude in SET are obtained by multi-ridge detection algorithm as reported in literature [58]. Meanwhile, we compare the results of proposed method when KF is activated and deactivated to show the performance of KF. Figs.7 and 8 plot the relative error of analysis results under the two noise levels. The relative error is defined as

$$\text{Relative error} = \frac{|\text{Estimated value} - \text{True value}|}{\text{True value}} \times 100\%. \tag{36}$$

Fig. 7 shows the analysis results corresponding to the case of low Gaussian white noise. In this situation, SET enjoys more accurate frequency and amplitude identifications in most times. For the 1st component, its frequency changes according to the sinusoidal law with a period of 8s. It changes by 1/2 circle in a computing frame (N=200). In this case where the frequency varies rapidly, the assumption of slow change is not satisfied, neither in our method nor in SET, so the errors are relatively high. For the 2nd component, because the SNR decreases with the attenuation of amplitude, the error increases gradually. For the 3rd component, when the amplitude changes suddenly, the amplitude error increases significantly.

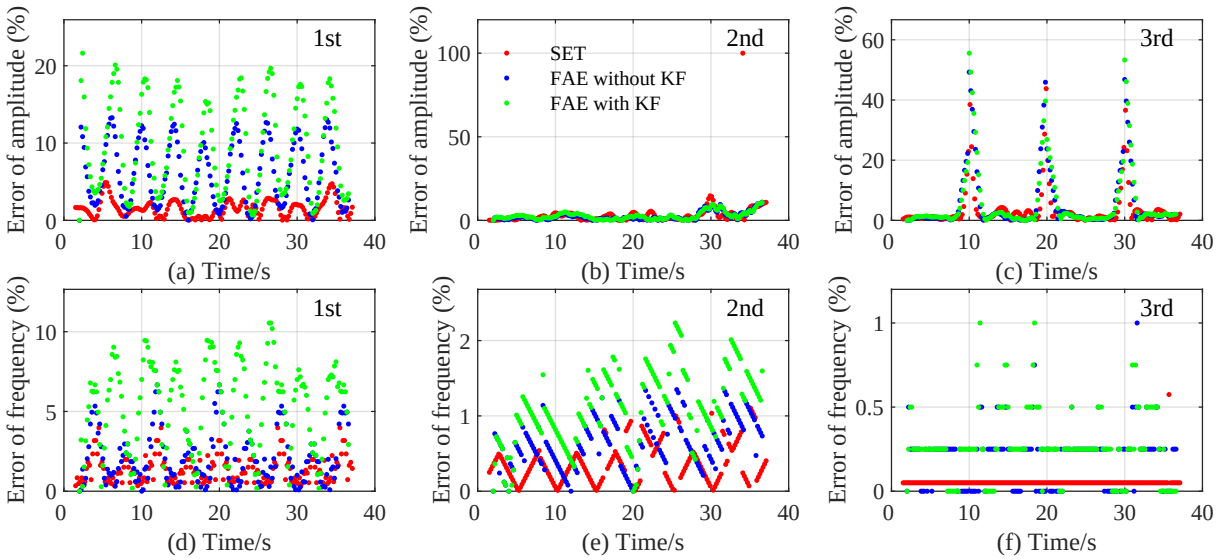

Figure 7: Analysis results of signal in Eq. (34) with a low noise level (σ_n=0.5, SNR=21.52dB). (a) $\sim$ (c) Error of 1st $\sim$ 3rd amplitude.(d) $\sim$ (f) Error of 1st $\sim$ 3rd frequency.

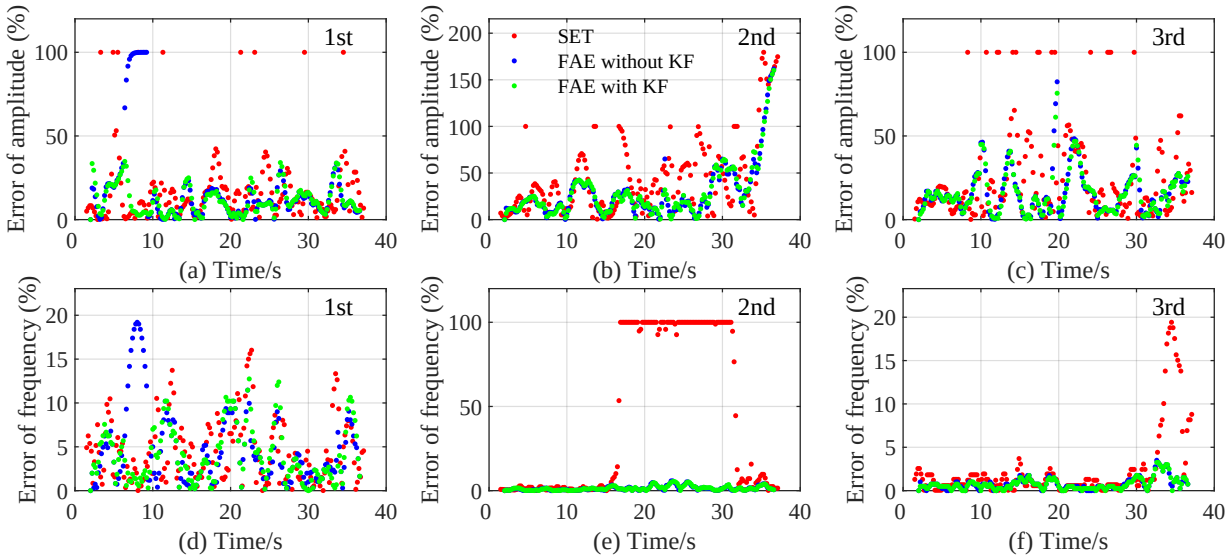

Figure 8: Analysis results of signal in Eq. (34) with a high noise level (σ_n=2.0, SNR=-2.57dB). (a) $\sim$ (c) Error of 1st $\sim$ 3rd amplitude. (d) $\sim$ (f) Error of 1st $\sim$ 3rd frequency.

However, at the high noise level, as shown in Fig. 8, our method exhibits better tracking accuracy and stability. For example, the 2nd and 3rd frequencies obtained by SET are much worse than the proposed method. This can be explained intuitively by the corresponding TF spectrogram in Fig. 9. The TFR obtained by SET are blurrier resulting in the difficulty in ridge detection. While the trajectories in the sparse TFR obtained by FAE (see in Fig.5(b)) are clearer, which is benefit to frequency extraction. What's more, it's found that the adoption of KF increase the robustness of the results, e.g., the frequency and amplitude in

Figs. 8(a) and (d).

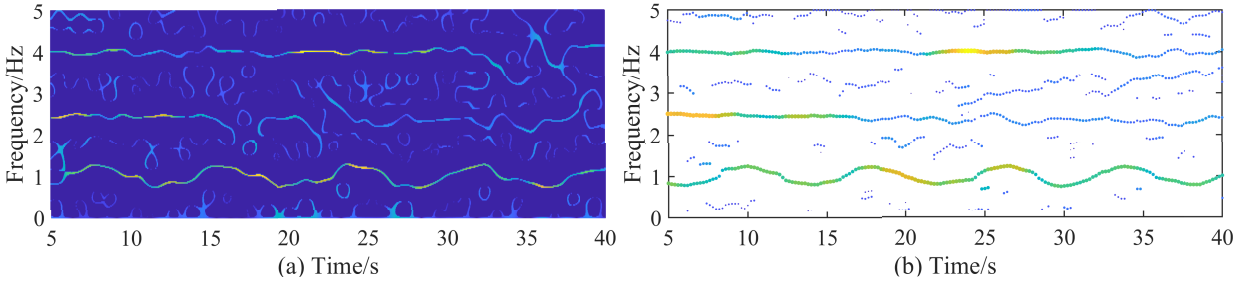

Figure 9: (a) The TFR obtained by SET. (b) The sparse TFR obtained by FAE proposed.

Although the method of SET can achieve online analysis as the manner we illustrated in Fig. 1. When we dive into more details, the differents between it and the proposed method come out. In the latter, frequency and amplitude are extracted from the spectra by APES and FSP. The signal for spectral analysis every time is N length and there is only one frequency and one amplitude to output for each order. While SET belongs to TF analysis methods, the frequency and amplitude are extracted from the spectrogram. The signal for TF analysis every time is $N+kS$ (N is the window's size, S is the moving spacing of window) length and there are $(k+1)$ frequencies and $(k+1)$ amplitudes to output for each order. This could be seen as online mini-batch processing of streaming data. Obviously, the method based on spectrum is more real-time with smaller analysis-lag.

The values of algorithms' parameters in the above numerical cases are summarized in Table B.1.

5. Realistic application

In this section, we will apply the proposed method to track the natural frequency and amplitude from the vibration signals of three actual structures recorded by their SHM systems. They are the ambient vibrations of a cable of a cable-stayed bridge (Section 5.1), a hanger of a half-through arch bridge (Section 5.2) and the girder of a suspension bridge (Section 5.3), as shown in Fig. 10. In each case, the analyzed signal lasts for 24 hours. The parameter settings in the algorithms are also summarized in Table B.1 and will not be described at the time of discussing these cases in detail.

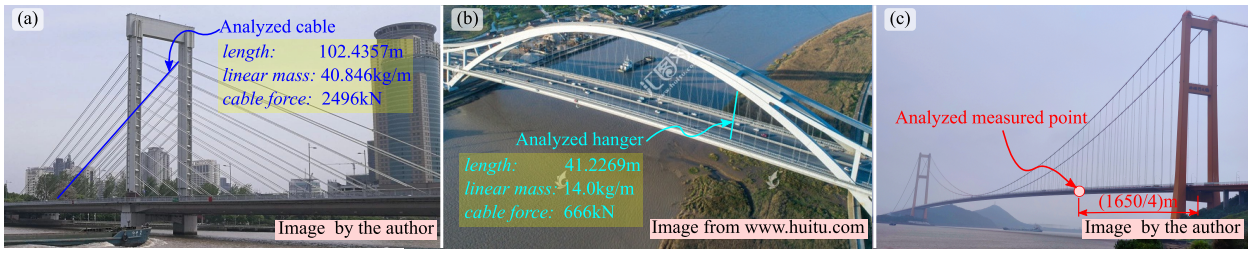

Figure 10: Actual structures that generate the vibrations. (a) A cable of a cable-stayed bridge, span layout: (105+97)m. (b) A hanger of a half-through arch bridge, span layout: (100+450+100)m. (c) The girder of a suspension bridge, span layout: (578+1650)m.

5.1. Cable vibrations of a cable-stayed bridge

The vibration signals of a cable of a cable-stayed bridge (Fig. 10(a)) that last from 0:00 at midnight to the next day (Fig. 11(a)) is analyzed. The cable is 102.4357m long with a linear mass of 40.846kg/m, and the cable force measured once is 2496kN. The first 6 orders of frequencies and amplitudes of interests are tracked by the proposed method.

Fig. 11(b) plots the time-varying frequencies, and the results during 2.5∼2.52h and 8.5∼8.52h are zoomed in Figs. 11(c) and (d). The signals corresponding to these two intervals represent the situations of relatively small and large vibrations, respectively. For the time between 2.5∼2.52h at night, there is supposed to be less traffic on the bridge deck than the time between 8.5∼8.52h at day, so the signal during the former interval has lower vibration energy. As we can see from Figs. 11(c) and (d), the proposed method is able to track the frequency continuously no matter under the low vibration energy or high. Actually, there still exist instants corresponding to some computing frames, where the shape of amplitude spectrum obtained by APES is not regular enough to exactly extract the natural frequencies and amplitudes of interest. Here a regular spectral shape means that the lobes at all the natural frequencies are higher than the lobes caused by noise. And the non-regular spectral shape is not only going to appear in the case of weak environmental excitation, but also in the case of high environmental excitation because not all modes are excited well. The non-regular spectral shape results in the raw extracted parameters not-aligned and then the frequency-alignment in Eq. (33) is activated to make the identified frequencies aligned and continuous. In the operation of frequency-alignment, if there are no candidates within the allowable variation range of reference value, the frequency in current

step is inherited from the last step while the amplitude is taken as half of value in the last step. Thus, in Figs. 11(c) and (d), some straight segments with amplitude close to zero are observable. These frequencies with near zero amplitudes indicate the corresponding modes are not well excited by the ambient. And we lost the tracking of the changing frequency during these periods since they are copied from the last step. Once the mode is excited by the ambient again, the method will recover the tracking of it.

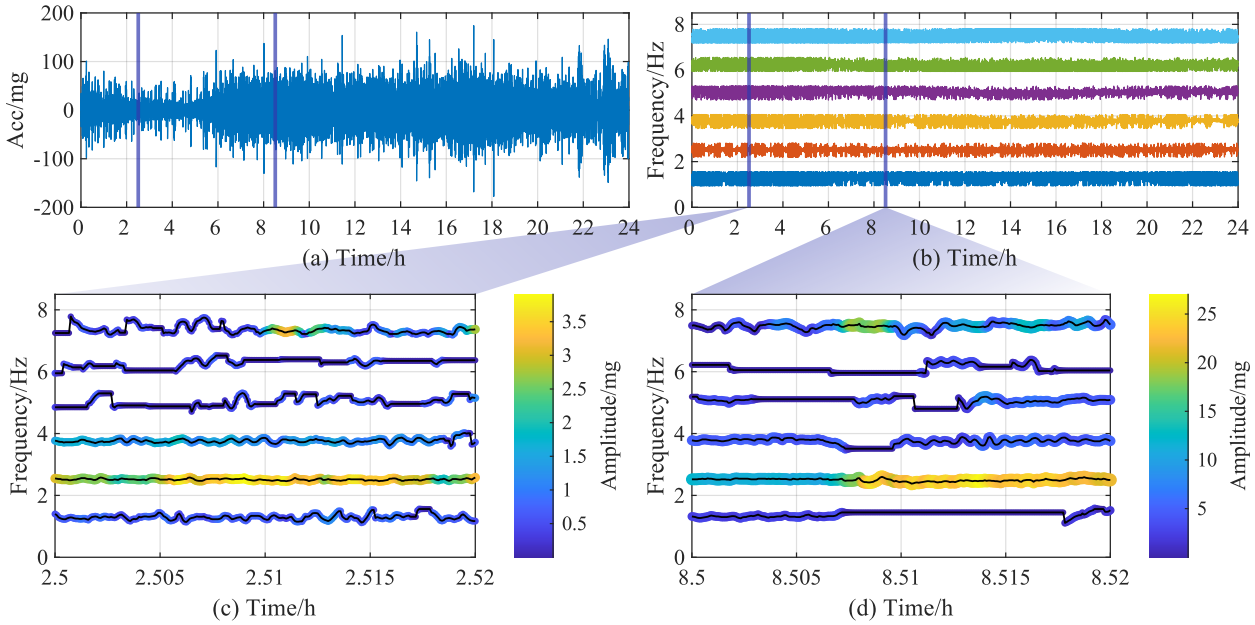

Figure 11: (a) Acceleration (Acc.) time history of a cable of a cable-stayed bridge in 24 hours. (b) Tracked first six natural frequencies. (c) Enlargement of (b) during 2.5∼2.52h . (d) Enlargement of (b) during 8.5∼8.52h.

Furthermore, Figs. 12 and 13 redraw the identified results in Figs. 11(c) and (d) in different panels separately according to the order, which provide a clearer view revealing the frequency fluctuations with respect to magnitudes of the identified amplitude. It's easy to find that, if the amplitude enjoys a continuous and moderately large magnitude (Figs. 12(b) and 13(b)), the corresponding frequency enjoys a smaller variance. Otherwise, if the amplitude is small and changes quickly (Figs. 12(a), (f) and 13(f)), the corresponding frequency is more discrete. This is reasonable since our method assumes that the signal during the analytical time window of N length is stationary. So, the relatively large variation of frequency may be more likely to be caused by the fast or abruptly changing amplitude, not the variation of structural characteristic itself. This also suggests that the frequency

results with large amplitudes (i.e., dominant modes) are more reliable.

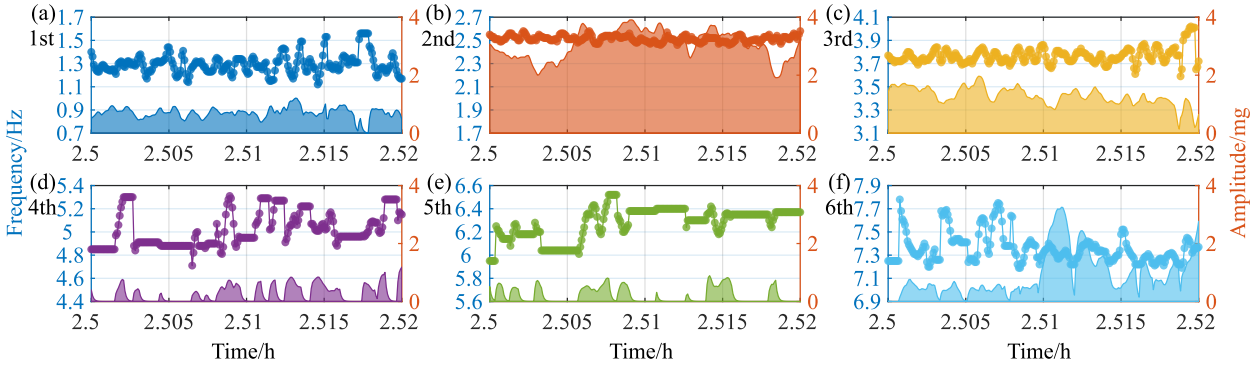

Figure 12: Frequency and amplitude of each order corresponding to the result in Fig. 11(c). (a)∼(f) Results from 1st to 6th order.

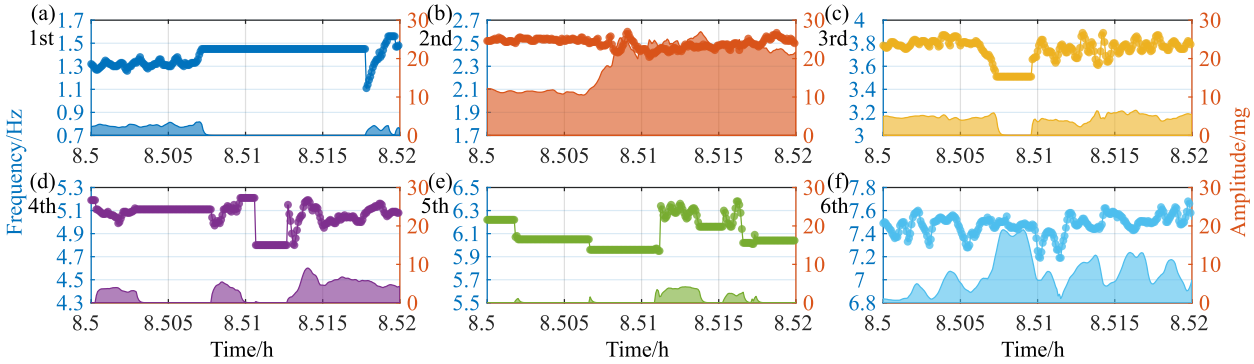

Figure 13: Frequency and amplitude of each order corresponding to the result in Fig. 11(d). (a)∼(f) Results from 1st to 6th order.

5.2. Hanger vibrations of a half-through arch bridge

In the last subsection, we have discussed the results of a cable vibration in detail. Here we give the analysis results of the 24-hour vibrations of a hanger of a half-through arch bridge (Fig. 10(b)) in brief since the major conclusions are similar. The hanger is 41.2269m long with a linear mass of 14.0kg/m, and the cable force measured once is 666kN.

Fig. 14(a) is the time history of the hanger vibrations and Fig. 14(b) shows the obtained frequencies of the first 4 modes. For a clearer view of the frequency changes, an enlargement of Fig. 14(b) in the time interval between 2.5h to 2.52h is given in Fig. 14(c). What's more, as what Figs. 12 and 13 do, Fig. 14(d) plots the frequency as well as the amplitude of each order separately, note that the limitations of the frequency and amplitude axes of each panel have been set to be consistent for comparison purpose.

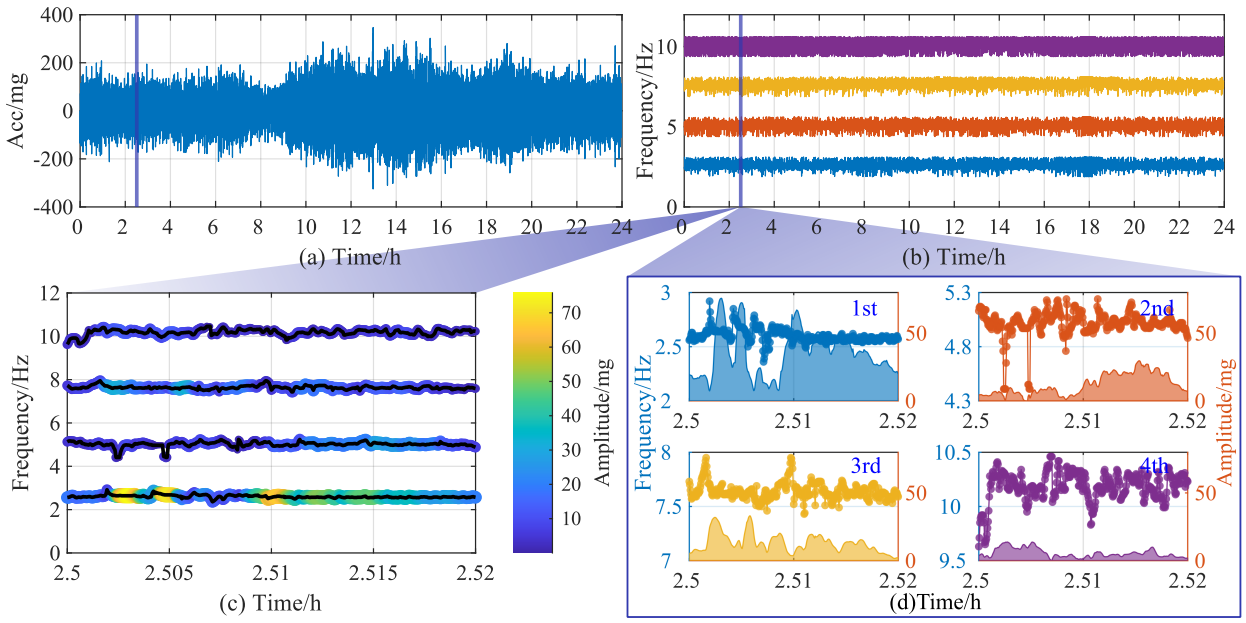

Figure 14: (a) Acceleration time history of a hanger of a half-through arch bridge in 24 hours. (b) Tracked first four natural frequencies. (c) Enlargement of (b) during 2.5∼2.52h. (d) Frequency and amplitude of each order corresponding to the result in (c).

As we can see, for this about 41m hanger, which is more well excited than the cable in the last subsection (compare Fig. 14(a) with 11(a)), showing more convincing frequency identification results (compare Fig. 14(c) with 11(c)). And Fig. 14(d) provides more proof that the frequency variation may also be caused by the rapidly varying amplitude not only by the real variation of itself. And the former is attributed to the assumption that the amplitude is slowly varying in an analytical window used in spectral analysis. And this assumption is common in many existing methods no matter whether to mode decomposition or TFA.

5.3. Girder vibrations of a suspension bridge

Lastly, we analyze the vibration signals generated by the girder of a suspension bridge, and the signals are recorded by an accelerometer installed at 1/4 of the main span, as shown in Fig. 10(c). Compared with the frequency and amplitude tracking of the cable or hanger, whose vibrational modes are well separated by almost equal distances, the same objective for the long-span suspension bridge becomes more challenging because its dominated modes are concentrated in the low frequency region and it's possible to happen mode crossover.

Similarly, the acceleration time history of the main girder is depicted in Fig. 15(a). And Fig. 15(b) is a scatter graph of the tracked results, showing the frequencies at different

instants and the brighter color means larger amplitude corresponded. As we can see, the suspension bridge exhibits 8 close vertical modes under 0.5Hz, and the frequency trajectories fluctuate around the early measured natural frequencies, which can be obtained from the Table 1 of the paper by H. Li et, al. [59]. Also, we list the 8 measured natural frequencies here. They are 0.0953, 0.1328, 0.1825, 0.2301, 0.2757, 0.3237, 0.4351 and 0.4918Hz. Observe the interval marked in Fig. 15, it is found that the bridge experienced vortex-induced vibrations (VIVs). The occurrence of VIV can be observed from both the acceleration time history in Fig. 15(a) and the extracted frequencies and amplitudes in Fig. 15(b). And the latter gives us more information about the energy distribution among different modes and which mode is participating in the VIV.

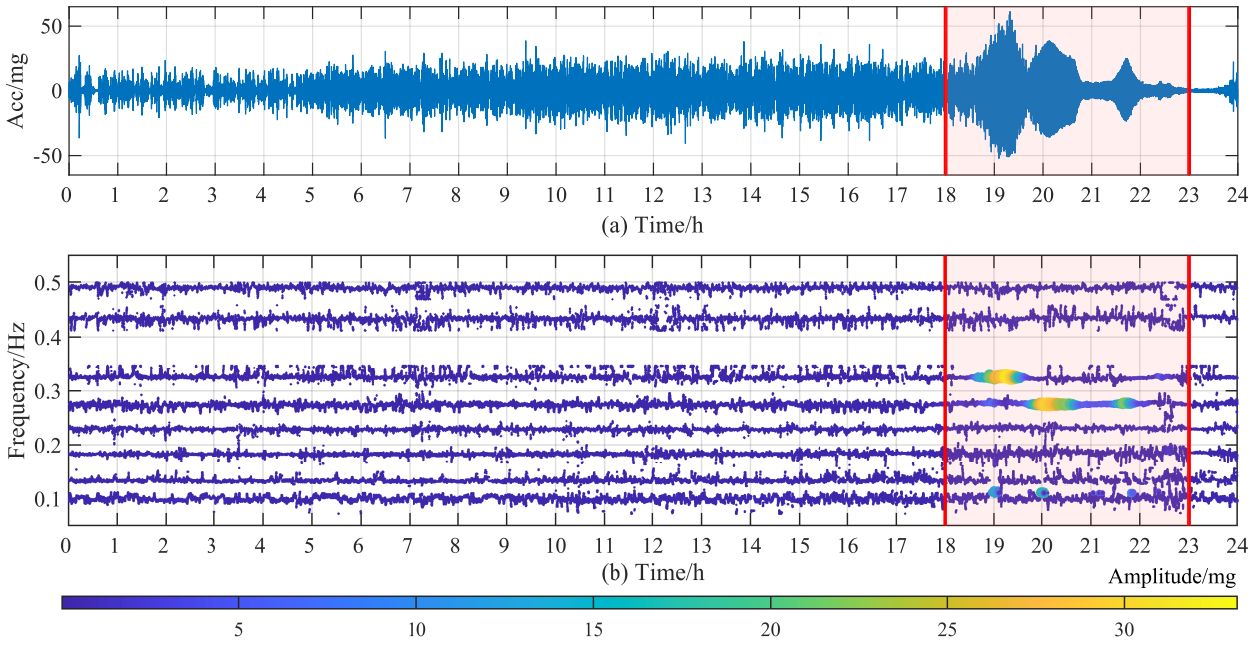

Figure 15: (a) Acceleration time history of the girder of a suspension bridge in 24 hours. (b) Tracked results in the low frequency zone.

We enlarge the results during the VIV as shown in Fig. 16(a) and plot the time-varying amplitude in Fig. 16(b). It's interesting to find that there exists an interchange for the modes participating in the VIV. At the beginning, the vibration energy is concentrated on the 6th mode, then the 6th mode's energy is progressively reduced while the 5th mode's energy is steadily raised until the 5th mode becomes the dominant one. Similarly, at the end of the disappearance of VIV on the 5th mode, the vibration energy is concentrated

on the 6th mode again for a short while. This should be the product of the interactions between structure and wind. Since VIV is generated when the frequency of vortex shedding caused by wind flow passing through the bridge section is close to a certain natural frequency of the structure under specific wind field conditions, this mode conversion may be due to the change of vortex shedding frequency caused by the change of wind field conditions. Additionally, from Fig. 16, we can determine when the VIV happens, which mode is most exciting and how long the VIV lasts. These discoveries are going to offer a new fundamental understanding of VIV of large-scale suspension bridge for the monitoring, perception and vibration control. Of course, this is beyond the scope of this paper and will be one of the focuses of our future work.

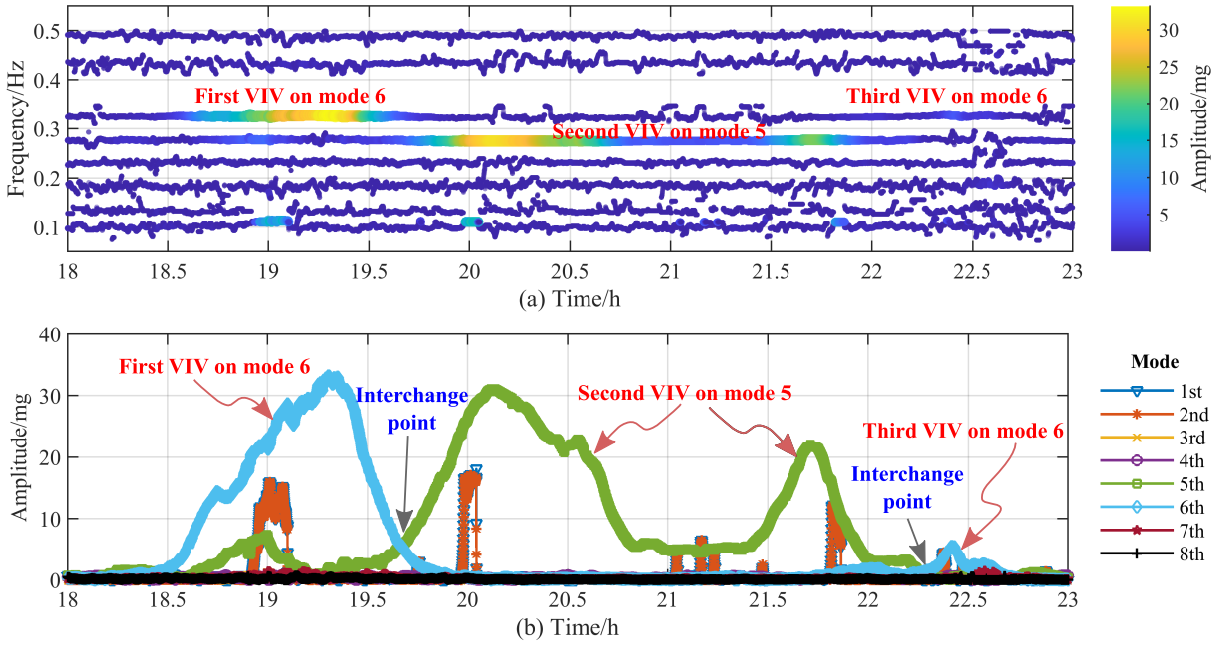

Figure 16: (a) Enlargement of the results in Fig. 16(b) during 18∼23h where VIVs occurred. (b) The tracked amplitudes corresponding to the first 8 mode during 18∼23h.

Furthermore, we pull our eyes back to the tracking results of the time-varying frequency (Figs. 15(b) and 15(a)). The proposed method is able to obtain the time series of frequency and amplitude of each mode. However, for this main girder with close modes in the low frequency zone, under environmental excitation, the mode mixing or coinciding are inevitable in the identified frequencies. Thus, the frequency trajectories in Figs. 15(b) and 15(a) at some intervals appear turbulent. On the other hand, to what extent this 'turbulence' is a

true reflection of structural characteristics is worth further investigation. This phenomenon is slighter in the cases of cable and hanger discussed above because their modes are more separated.

6. Conclusion

For the purpose of tracking natural frequency and amplitude of civil structures from their monitoring data, this paper develops a method based on APES spectrum postprocessing and KF. Its effectiveness is validated through the analysis of numerical examples and the comparison with SET, exhibiting more accurate and robust results in the case of low SNR. What's more, we employ the proposed method to analyze the vibration signals generated by actual engineering structures. While proving its ability to extract frequency and amplitude online in environmental vibrations, some main conclusions are reached as follows:

1. Under the random ambient excitation, the magnitude and changing rate of amplitude of different modes are quite different. For modes with large and slowly changed amplitude, the variance of identified frequency is small (Fig. 13(b)), and for modes with small and fast changed amplitude (not well excited), the variance of identified frequency is large (Fig. 13(f)). This is because the severe amplitude modulation is going to submerge the information of structural oscillations and bring errors in the identified frequency positions. On the other hand, those frequencies with relatively large and slow-changing amplitudes are more reliable. In other words, the identified amplitude can be used as a measure of the reliability of identified frequency.
2. Under environmental excitation, it is uncertain which modes are well excited and the SNR will be quite low sometimes resulting the characteristic information of vibrations immerged in noises. This brings challenges to the online tracking of frequency and amplitude. The proposed method utilizes KF and frequency alignment scheme to keep the robustness of the tracking results and it shows positive performance in long-term analysis. However, for the modes that are very closely spaced, the tracking results may be mixed in the case of large frequency variation is caused by the varying ambient.

3. The frequency alignment scheme provided in this paper is adaptive to separated modes, which is true for most engineering structures. When handling with the situation of mode crossover, it has to be newly designed.

Our future work will be focused on the more applications of the proposed method, such as the damage diagnosis and abnormal event detection, e.g., the VIV monitoring of suspension bridge discussed in the Section 5.3.

Appendix A. Discussion on parameter setting in Algorithms 1∼4

Appendix A.1. Parameters in Algorithm 1: APES

a. Computing frame length N should be selected according to the longest period of signal components, and N is recommended to be 5 to 10 times the longest period.

b. Calculating frequency $\boldsymbol{\omega} = [\omega_1, \omega_2, \cdots, \omega_K]^T$ is assigned according to frequency interval of interest, and $0 \leq \omega_i \leq F_s/2$. Generally, the elements of frequency vector $\boldsymbol{\omega}$ are uniformly distributed, and the spacing $\Delta\omega := \omega_{k+1} - \omega_k$ is set according the requirement of precision, such as 0.01Hz, 0.001Hz.

c. Filter order M should be selected as a trade-off between the resolution and the statistical stability, and it is suggested $N/4 \leq M \leq N/2$ [52, 60].

Appendix A.2. Parameters in Algorithm 2: FSP

a. Order m (Eq. (17)) is used to take the mth power of the normalized amplitude, which is beneficial to promote convergence and effectiveness of frequency squeezing, and is suggested as $30 \sim 60$.

b. Step size P (Eq. (18)) is a key parameter for frequency squeezing. A large value of P is helpful for spectrum concentration. However, a too large P will lead to the missing of natural frequency with relatively small amplitude. Generally, the frequency span composed of the $2P + 1$ spectral lines shouldn't include more than 2 natural frequencies.

c. Threshold δ_ω (Eq. (19)) is used as a criterion to judge the edges of frequency clusters. Based on the observation that frequency points will be extremely gathered at the natural frequencies after several steps of frequency squeezing (Eq. 18), a small δ_ω should be selected and we recommend it as $\delta_\omega = 0.01\Delta\omega$, where $\Delta\omega := \omega_{i+1} - \omega_i$.

d. Threshold δ_s is used to stop iteration of frequency squeezing, a small value such as 1e-4∼1e-8 could be set.

e. Maximum iterations I is for compulsively stopping the iteration in case the frequency squeezing fails to converge in finite steps. In this paper, it is set as $I = 50$.

Appendix A.3. Parameters in Algorithm 3: Automatic frequency and amplitude extraction

a. Threshold δ_Ω (Eq. (20)) is used to separate the processed frequency $\hat{\boldsymbol{\omega}}$ into P sub-intervals in which the maximum distance between individuals is less than δ_Ω. And we recommend it as F_s/N, which is the precision of fast Fourier Transform (FFT).

b. Order n (Eq. (24a)) is used to truncate decimal places, and it's suggested as $n = \max(1, \lfloor \log_{10}(1/\Delta\omega) \rfloor)$.

c. δ_d (Eq. (22)) determines a threshold to judge if the frequency deviation caused by FSP is acceptable, and it's also suggested to be F_s/N.

Appendix A.4. Parameters in Algorithm 4: Frequency and amplitude tracking using APES postprocessing and KF

a. Initial frequency $\boldsymbol{\omega}_{n,0}$ and amplitude $\boldsymbol{\alpha}_{n,0}$ can be obtained from historical data, and $\boldsymbol{\omega}_{n,0}$ can also be assigned as the reference frequency $\boldsymbol{\omega}_r$.

b. Updating block length S controls the output frequency of results, i.e., a smaller S means a higher time resolution. Meanwhile, the computations in a single step should be completed within S samples, so S can't be too small (see in Eq. 8). And we recommended it as 1/10 to 1/5 times the computing frame length N.

c. Weight ρ (Eqs. (31f), (31g) balances the energy metric and distance metric for updating covariances of process noise and measurement noise. If we emphasize the energy

metric (more faith in prediction), set a large ρ; if we emphasize the distance metric (more faith in measurement), set a small ρ. And it's 0.5 used in this paper.

d. $\boldsymbol{\delta}_a$ is the allowable range of frequency variation with respect to reference value $\boldsymbol{\omega}_r$. It could be a vector with the same length as the reference frequency vector $\boldsymbol{\omega}_r$, that is, each order of reference frequency $\omega_{r,i}$ has its own allowable variation range; or it could be a scalar, that is, all order reference frequencies have the same allowable variation range.

Appendix B. Specific parameter setting in the numerical and realistic cases

Table B.1: Parameter setting of the proposed method in analyzing numerical and realistic cases.

Alg.	Para.	Sec. 4	Sec. 5.1	Sec. 5.2	Sec. 5.3
1	N	200	150	100	4000
	$\boldsymbol{\omega}$	$[0, 0.01, \cdots, 5]$	$[0, 0.01, \cdots, 8]$	$[0, 0.01, \cdots, 12]$	$[0, 0.001, \cdots, 0.5]$
	M	33	37	37	666
	F_s	50	50	50	50
2	m	50	50	50	50
	P	25	60	100	20
	δ_ω	1e-4	1e-4	1e-4	1e-5
	δ_s	1e-8	1e-6	1e-6	1e-6
	I	50	50	50	50
3	n	2	2	2	3
	δ_Ω	0.25	0.3333	0.5	0.0125
	δ_d	0.25	0.3333	0.5	0.0125
4	S	10	15	10	100
	$\boldsymbol{\omega}_r$	$\boldsymbol{\omega}_{n,k}$	1.25[1,2,3,4,5,6]	2.50[1,2,3,4]	[.095,.133,.183,.230, .275,.324,.435,.492]
	$\boldsymbol{\omega}_{n,0}$	[1.1,2.49,4.0]	1.25[1,2,3,4,5,6]	2.50[1,2,3,4]	[.095,.133,.183,.230, .275,.324,.435,.492]
	$\boldsymbol{\alpha}_{n,0}$	[3.0,3.71,3.0]	[1,1,1,1,1,1]	[1,1,1,1]	[1,1,1,1,1,1,1,1]
	ρ	0.5	0.5	0.5	0.5
	δ_a	0.1	0.3125	0.625	0.0238

Acknowledgments

This work is supported by the Opening Foundation of National Key Laboratory of Bridge Structural Health and Safety; the National Nature Science Foundation of China (Grant No. 5187849); the National Key R&D Program of China (2017YFF0205605).

References

[1] Y. Bao, Z. Chen, S. Wei, Y. Xu, Z. Tang, H. Li, The state of the art of data science and engineering in structural health monitoring, Engineering 5 (2019) 234–242.

[2] S. Sony, S. Laventure, A. Sadhu, A literature review of next-generation smart sensing technology in structural health monitoring, Structural Control and Health Monitoring 26 (2019) e2321.

[3] B. Boashash, Estimating and interpreting the instantaneous frequency of a signal. i. fundamentals, Proceedings of the IEEE 80 (1992) 520–538.

[4] O. Salawu, Detection of structural damage through changes in frequency: a review, Engineering structures 19 (1997) 718–723.

[5] M. Mousavi, D. Holloway, J. Olivier, A. H. Gandomi, Beam damage detection using synchronisation of peaks in instantaneous frequency and amplitude of vibration data, Measurement 168 (2021) 108297.

[6] H. Pan, M. Azimi, F. Yan, Z. Lin, Time-frequency-based data-driven structural diagnosis and damage detection for cable-stayed bridges, Journal of Bridge Engineering 23 (2018) 04018033.

[7] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, H. H. Liu, The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis, Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences 454 (1998) 903–995.

[8] G. Chen, Z. Wang, A signal decomposition theorem with hilbert transform and its application to narrowband time series with closely spaced frequency components, Mechanical systems and signal processing 28 (2012) 258–279.

[9] J. Gilles, Empirical wavelet transform, IEEE transactions on signal processing 61 (2013) 3999–4010.

[10] K. Dragomiretskiy, D. Zosso, Variational mode decomposition, IEEE transactions on signal processing 62 (2013) 531–544.

[11] Z. Feng, D. Zhang, M. J. Zuo, Adaptive mode decomposition methods and their applications in signal analysis for machinery fault diagnosis: a review with examples, IEEE access 5 (2017) 24301–24331.

[12] T. Liu, Z. Luo, J. Huang, S. Yan, A comparative study of four kinds of adaptive decomposition algorithms and their applications, Sensors 18 (2018) 2120.

[13] M. Barbosh, P. Singh, A. Sadhu, Empirical mode decomposition and its variants: a review with applications in structural health monitoring, Smart Materials and Structures 29 (2020) 093001.

[14] M. Civera, C. Surace, A comparative analysis of signal decomposition techniques for structural health monitoring on an experimental benchmark, Sensors 21 (2021) 1825.

[15] M. Feldman, Hilbert transform in vibration analysis, Mechanical systems and signal processing 25 (2011) 735–802.

[16] A. Baccigalupi, A. Liccardo, The huang hilbert transform for evaluating the instantaneous frequency evolution of transient signals in non-linear systems, Measurement 86 (2016) 1–13.

[17] P. Ni, J. Li, H. Hao, Y. Xia, X. Wang, J.-M. Lee, K.-H. Jung, Time-varying system identification using variational mode decomposition, Structural Control and Health Monitoring 25 (2018) e2175.

[18] Y. Xin, J. Li, H. Hao, Enhanced vibration decomposition method based on multisynchrosqueezing transform and analytical mode decomposition, Structural Control and Health Monitoring 28 (2021) e2730.

[19] P. F. Pai, Online tracking of instantaneous frequency and amplitude of dynamical system response, Mechanical Systems and Signal Processing 24 (2010) 1007–1024.

[20] R. Zhong, P. F. Pai, An instantaneous frequency analysis method of stay cables, Journal of Low Frequency Noise, Vibration and Active Control 40 (2021) 263–277.

[21] A. Cicone, J. Liu, H. Zhou, Adaptive local iterative filtering for signal decomposition and instantaneous frequency analysis, Applied and Computational Harmonic Analysis 41 (2016) 384–411.

[22] D. Vakman, On the analytic signal, the teager-kaiser energy algorithm, and other methods for defining amplitude and frequency, IEEE Transactions on signal processing 44 (1996) 791–797.

[23] Y. Bao, Z. Shi, J. L. Beck, H. Li, T. Y. Hou, Identification of time-varying cable tension forces based on adaptive sparse time-frequency analysis of cable vibrations, Structural Control and Health Monitoring 24 (2017) e1889.

[24] X. Zhang, J. Peng, M. Cao, D. Damjanović, W. Ostachowicz, Identification of instantaneous tension of bridge cables from dynamic responses: Strict algorithm and applications, Mechanical Systems and Signal Processing 142 (2020) 106729.

[25] F. Hlawatsch, F. Auger, Time-frequency analysis, John Wiley & Sons, 2013.

[26] L. Stanković, I. Djurović, S. Stanković, M. Simeunović, S. Djukanović, M. Daković, Instantaneous frequency in time–frequency analysis: Enhanced concepts and performance of estimation algorithms, Digital Signal Processing 35 (2014) 1–13.

[27] S. Mann, S. Haykin, The chirplet transform: Physical considerations, IEEE Transactions on Signal Processing 43 (1995) 2745–2761.

[28] Z. Peng, G. Meng, F. Chu, Z. Lang, W. Zhang, Y. Yang, Polynomial chirplet transform with application to instantaneous frequency estimation, IEEE Transactions on Instrumentation and Measurement 60 (2011) 3222–3229.

[29] F. Auger, P. Flandrin, Improving the readability of time-frequency and time-scale

representations by the reassignment method, IEEE Transactions on signal processing 43 (1995) 1068–1089.

[30] I. Daubechies, J. Lu, H.-T. Wu, Synchrosqueezed wavelet transforms: An empirical mode decomposition-like tool, Applied and computational harmonic analysis 30 (2011) 243–261.

[31] G. Yu, M. Yu, C. Xu, Synchroextracting transform, IEEE Transactions on Industrial Electronics 64 (2017) 8042–8054.

[32] B. Boashash, Estimating and interpreting the instantaneous frequency of a signal. ii. algorithms and applications, Proceedings of the IEEE 80 (1992) 540–568.

[33] V. Popović-Bugarin, S. Djukanović, Efficient instantaneous frequency estimation in high noise based on the wigner distribution, Signal Processing 157 (2019) 25–29.

[34] R. A. Carmona, W. L. Hwang, B. Torrésani, Multiridge detection and time-frequency reconstruction, IEEE transactions on signal processing 47 (1999) 480–492.

[35] S. Meignen, D.-H. Pham, S. McLaughlin, On demodulation, ridge detection, and synchrosqueezing for multicomponent signals, IEEE Transactions on Signal Processing 65 (2017) 2093–2103.

[36] C. Wang, W.-X. Ren, Z.-C. Wang, H.-P. Zhu, Instantaneous frequency identification of time-varying structures by continuous wavelet transform, Engineering Structures 52 (2013) 17–25.

[37] D. Iatsenko, P. V. McClintock, A. Stefanovska, Extraction of instantaneous frequencies from ridges in time–frequency representations of signals, Signal Processing 125 (2016) 290–303.

[38] L. Rankine, M. Mesbah, B. Boashash, If estimation for multicomponent signals using image processing techniques in the time–frequency domain, Signal Processing 87 (2007) 1234–1250.

[39] J. Liu, X. Wei, R.-H. Qiu, J.-Y. Zheng, Y.-J. Zhu, I. Laory, Instantaneous frequency extraction in time-varying structures using a maximum gradient method, Smart Struct. Syst 22 (2018) 359–368.

[40] M. Brajović, V. Popović-Bugarin, I. Djurović, S. Djukanović, Post-processing of time-frequency representations in instantaneous frequency estimation based on ant colony optimization, Signal Processing 138 (2017) 195–210.

[41] C. Wang, J. Zhang, H. P. Zhu, A combined method for time-varying parameter identification based on variational mode decomposition and generalized morse wavelet, International Journal of Structural Stability and Dynamics 20 (2020) 2050077.

[42] Y. Kaya, E. Safak, Real-time analysis and interpretation of continuous data from structural health monitoring (shm) systems, Bulletin of Earthquake Engineering 13 (2015) 917–934.

[43] S. Quqa, L. Landi, P. P. Diotallevi, Instantaneous modal identification under varying structural characteristics: A decentralized algorithm, Mechanical Systems and Signal Processing 142 (2020) 106750.

[44] B. Bhowmik, M. Krishnan, B. Hazra, V. Pakrashi, Real-time unified single-and multi-channel structural damage detection using recursive singular spectrum analysis, Structural Health Monitoring 18 (2019) 563–589.

[45] B. Bhowmik, T. Tripura, B. Hazra, V. Pakrashi, Real time structural modal identification using recursive canonical correlation analysis and application towards online structural damage detection, Journal of Sound and Vibration 468 (2020) 115101.

[46] M. Krishnan, B. Bhowmik, B. Hazra, V. Pakrashi, Real time damage detection using recursive principal components and time varying auto-regressive modeling, Mechanical Systems and Signal Processing 101 (2018) 549–574.

[47] R. Soman, P. Malinowski, K. Majewska, M. Mieloszyk, W. Ostachowicz, Kalman filter based neutral axis tracking in composites under varying temperature conditions, Mechanical Systems and Signal Processing 110 (2018) 485–498.

[48] R. Soman, W. Ostachowicz, Kalman filter based neutral axis tracking for damage detection in composites structures under changing axial loading conditions, Composite Structures 206 (2018) 517–525.

[49] J. Li, P. Stoica, An adaptive filtering approach to spectral estimation and sar imaging, IEEE Transactions on Signal Processing 44 (1996) 1469–1484.

[50] L. Stanković, M. Daković, T. Thayaparan, A real-time time-frequency based instantaneous frequency estimator, Signal processing 93 (2013) 1392–1397.

[51] X. Yu, D. Dan, Block-wise recursive apes aided with frequency-squeezing postprocessing and the application in online analysis of vibration monitoring signals, Mechanical Systems and Signal Processing 162 (2022) 108063. doi:https://doi.org/10.1016/j.ymssp.2021.108063.

[52] E. G. Larsson, J. Li, P. Stoica, High-resolution nonparametric spectral analysis: Theory and applications, in: High-resolution and robust signal processing, volume 4, Marcel Dekker, 2003, pp. 153–253.

[53] P. Stoica, R. L. Moses, et al., Spectral analysis of signals, Pearson Prentice Hall Upper Saddle River, NJ, 2005.

[54] S. J. Julier, J. K. Uhlmann, Unscented filtering and nonlinear estimation, Proceedings of the IEEE 92 (2004) 401–422.

[55] S. J. Julier, J. K. Uhlmann, New extension of the kalman filter to nonlinear systems, in: Signal processing, sensor fusion, and target recognition VI, volume 3068, International Society for Optics and Photonics, 1997, pp. 182–193.

[56] R. Van Der Merwe, et al., Sigma-point Kalman filters for probabilistic inference in dynamic state-space models, Ph.D. thesis, OGI School of Science & Engineering at OHSU, 2004.

[57] V. Kučera, The discrete riccati equation of optimal control, Kybernetika 8 (1972) 430–447.

[58] J. Li, X. Zhu, S.-s. Law, B. Samali, Time-varying characteristics of bridges under the passage of vehicles using synchroextracting transform, Mechanical Systems and Signal Processing 140 (2020) 106727.

[59] H. Li, S. Laima, J. Ou, X. Zhao, W. Zhou, Y. Yu, N. Li, Z. Liu, Investigation of vortex-induced vibration of a suspension bridge with two separated steel box girders based on field measurements, Engineering Structures 33 (2011) 1894–1907.

[60] P. Stoica, A. Jakobsson, J. Li, Matched-filter bank interpretation of some spectral estimators, Signal Processing 66 (1998) 45–59.