buildings

Article

Online Sifting Technique for Structural Health Monitoring Data Based on Recursive EMD Processing Framework

Danhui Dan [1,*], Chenqi Wang [1], Ruiyang Pan [1] and Yangmei Cao [2]

[1] School of Civil Engineering, Tongji University, 1239 Siping Road, Shanghai 200092, China
[2] China Railway Siyuan Survey and Design Group Co., Ltd., Wuhan 430063, China
* Correspondence: dandanhui@tongji.edu.cn; Tel.: +86-139-1807-5836

Abstract: Real-time and online screening techniques for single load effect signal monitoring are one of the key issues in smart structure monitoring. In this paper, an online signal sifting framework called online recursive empirical mode decomposition (EMD) is proposed. The framework is based on an improved EMD that optimizes the boundary effect by using extreme value recursion and eigensystem realization algorithm (ERA) extension, and combines the intrinsic mode functions (IMFs) correlation coefficient and adaptive filtering to select IMFs for signal reconstruction to achieve the sifting purpose. When applied to simulated signals, the method satisfies the requirements of signal sifting in an online environment with high adaptivity, low parameter sensitivity and good robustness. The method was applied to the dynamic strain data collected by the health monitoring system of Daishan Second Bridge to achieve real-time online sifting of strain signals caused by traffic loads, which provided the basis for subsequent data analysis applications and confirmed the value of the application in a real bridge health monitoring system.

Keywords: online signal sifting; online recursive EMD; intrinsic mode functions; ERA

Citation: Dan, D.; Wang, C.; Pan, R.; Cao, Y. Online Sifting Technique for Structural Health Monitoring Data Based on Recursive EMD Processing Framework. *Buildings* **2022**, *12*, 1312. https://doi.org/10.3390/buildings12091312

Academic Editors: Minshui Huang and Jianfeng Gu

Received: 25 July 2022
Accepted: 19 August 2022
Published: 26 August 2022

1. Introduction

The structural health monitoring system (SHM) has the feature of real-time and continuous acquisition of structural performance parameters compared with conventional structural detection, which enables online data processing and engineering structural health condition assessment. The condition assessment of engineering structures includes working condition assessment and service performance assessment; the former is mostly related to the working load effect, while the latter needs to extract the relevant indexes from the structural effect under working load. The actual structural effect signals collected by SHM are the result of mixing and superposition under multiple loads [1], from which further single working load effects need to be obtained. In order to fit the advantages of real-time signal processing of SHM, it is necessary to propose online real-time sifting techniques for individual load effects in a streaming data environment to finally realize real-time sensing of engineering structural health status.

In order to obtain the individual load effects, we need to first decompose the mixed effect monitoring signals collected by the sensors, and then reorganize them to obtain the subcomponents according to the time-frequency domain characteristics of the working load effects. For a bridge structure, the working load is the traffic vehicle load, and the induced monitoring signals have obvious differences in amplitude and frequency from other loads, which can be sifted by filters in the streaming data environment. Early stream monitoring data calculations mainly used linear filters, such as Wiener filtering [2], which are easy to design and implement. However, linear filters are not ideal for multi-source complex noise and sharp impulse signals [3]. Most engineering data have non-smooth and non-linear characteristics, resulting in limited applicability of linear filters, so more non-linear processing methods have been developed [4]. This approach includes wavelet thresholding, adaptive filtering, and online Kalman filtering [5,6]. The basic principle of wavelet thresholding is that wavelet coefficients smaller than a predetermined threshold are zero. It is proved that the online multiscale filtering method based on wavelet thresholding is significantly better than the linear filter for signal sifting, but the effect will be limited by the threshold parameter, especially for the dynamic online data with large changes or lacking pre-validation [7]. Adaptive filtering is widely used for real-time processing of various types of engineering data, such as noise reduction, peak picking, and extraction of specified waveforms [8,9]. However, the effect of this method is limited by the expected signal and the error criterion, resulting in poor applicability to dynamic online signals.

In summary, the online streaming processing method should have the following characteristics: (1) high adaptivity—data sifting from the perspective of data characteristics and excellent applicability to non-linear, non-smooth signals; (2) low dependence of preset parameters—few preset parameters and low sensitivity to accommodate online dynamic signals; (3) data articulation—streaming calculates uniformity and coherence between different data frame signals [10]. To satisfy the above requirements, we propose an online streaming signal processing framework based on an improved EMD method called online recursive EMD (OREMD).

The framework uses EMD as the basis for signal processing, which is a completely data-driven approach with no basis function requirements for online dynamic data that reflect the physical characteristics of the original time series signal [11,12]. The EMD decomposes the original signal into multiple IMFs with different frequency modes according to an intrinsic time scale [13]. The core of EMD signal sifting is that the effective information of the signal is concentrated in the low-frequency IMFs and decreases towards the high-frequency eigenmode IMFs, which has been proven to be used for signal sifting [14] and has been widely used in various fields in recent years [15–18]. This paper proposes a combination of correlation coefficient and adaptive filtering to select the effective IMFs, which further improves the applicability of the method.

The main problems with EMD are the boundary effects and modal mixing, which can generate boundary effect errors and propagate within the data, affecting the accuracy [13,19]. For online data, the boundary effects at both ends of the signal should be discussed separately due to the deterministic prior sequence data. Usually, the online signal for a certain frame can be divided into the initial boundary effect and the termination boundary effect. The issue of modal mixing will not affect the process in this paper and will not be discussed for the time being.

For the initial boundary effect, the traditional approach is overlapping the signal of the previous frame, which is not conducive to long-term online processing due to repeated calculations and time delays. This paper uses the known nature of the pre-order time-series data to optimize the initial boundary effect by transmitting the local extrema of the IMFs. It has been verified that this method can effectively improve the accuracy of the online calculation results. For the termination boundary effects, the common solution is to extend the original signal at the endpoints in order to fit the envelope outside the existing data range. Some methods apply data extension, including wave extension methods [13], and autoregressive moving average [20]. In order to further improve the efficiency, support vector machine prediction (SVR) [21], autoregressive (AR) based forecast endpoint extensions [22], and prediction via radial basis function (RBF) neural networks have been proposed [23]. The SVR uses the generalization ability of vector machines to predict extreme value points and has been applied in several fields [24,25]. Although SVR is an effective prediction method, non-stationary time series have a significant impact on its prediction accuracy [26]. The mirror extension method is simple to use and has a wide range of applications, but the number of levels of the extension data and the actual number of levels of the method will have deviations that affect the results. The neural network algorithm is tedious in the preliminary process. Considering the ease of use and stability of the online calculation method, this paper adopts the ERA extension method. The purpose of ERA is to provide more suitable data with the system parameters of the existing signal rather than identifying the system parameters. So, ERA extension as the main method and mirror extension as the alternative method are adopted.

In summary, this paper proposes an online recursive EMD processing framework (OREMD) based on EMD filtering with ERA extension and adopts a streaming computing model to update the processing data in real-time by data frame windows. The sifting results of this method are compared with the online median filter to demonstrate the effectiveness of OREMD and the low sensitivity of parameters. Finally, the method is applied to the screening of vehicle-caused loads in actual bridge projects, and the results show that it could effectively improve the data quality and reliability of subsequent judgments.

2. Recursive Empirical Modal Decomposition

2.1. Empirical Mode Decomposition (EMD)

Empirical mode decomposition (EMD) is an adaptive time-domain data analysis method applicable to non-linear and non-stationary signals [13]. The basic principle of EMD is that any signal can be decomposed into several intrinsic mode functions (IMFs). The IMFs obtained from the local time feature scales decomposition of the data have different frequency characteristics and satisfy the following two characteristics: (1) the number of extreme points and zero crossing points differ at most by one; (2) the average value of the upper envelope and the lower envelope is zero at any moment. Original signal $x(t)$ is finally decomposed as $x(t) = \sum_{i=1}^{n} imf_i + r_n$ after the sifting process. imf_i represents the ith IMF, and r_n is the residue.

The above processing is performed in offline batch processing and is not suitable for online monitoring real-time decomposition. In order to achieve continuous signal decomposition in an online monitoring environment, a natural approach is to frame block the streaming signal and then decompose it separately using classical EMD.

2.2. Online EMD Boundary Effect

Optimal EMD is batch processing of existing signals, but the online system signal volume increases with time, so each time the calculation repeats the previous process, resulting in increased processing costs with time and decreasing calculation efficiency. For the theoretical "infinite length" format of online signals, stream processing is used to process the online signals entering the system, which has high time-effect, low computing costs.

The accuracy of the stream processing is influenced by the boundary effects on both sides of the EMD of the currently calculated frame. According to this, the online calculation frame is divided into three parts: the initial boundary effects section, intermediate section, and termination boundary effects section (shown in Figure 1).

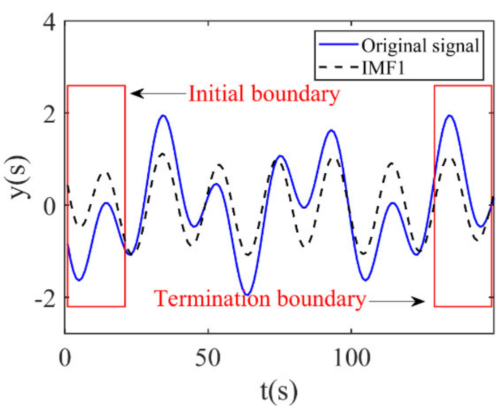


Figure 1. Boundary effect on both sides.

The initial boundary effect is often handled by signal overlap, which has two drawbacks. First, the overlap of signals implies the existence of a time delay effect. Second, the overlapped signal part increases the computational effort. For the termination boundary effect, the common method is extending the boundary, but the boundary effect still has not been solved [19]. In this paper, a combination of recursive transmission extreme and data extension methods are selected to meet the online EMD requirements.

2.3. Initial Boundary Effect: Online Recursive EMD Based on IMF Extreme Value Memory and Transfer

The key to solving the initial boundary effect problem is to provide more reasonable extension information. By passing the end extrema of the same-order IMF components generated in the previous frame to the next frame for iteration instead of the whole part of the signal, one can ensure that the envelope of the next frame iteration with the partial extrema of the previous frame still satisfies the IMF characteristics. On the one hand, it is beneficial to realize the seamless connection of the same-order IMF components of adjacent frame signals, and on the other hand, it reduces the amount of iterative calculation, so that the decomposition result is closer to the signal result of one batch processing.

$\{x(t_i)\}$ represents the data of ith frame; $\left\{u^k_{max}(t_i)\right\}$, $\left\{u^k_{min}(t_i)\right\}$ represent the final maximum values and minimum values of the kth IMF of the ith frame; $\{z_{max}[x(t_i)]\}$, $\{z_{min}[x(t_i)]\}$ represent the maximum and minimum values of the current data. The flow chart is shown in Figure 2, and the flow of the algorithm is shown in Algorithm 1.

Algorithm 1. Recursion EMD

Initialize $i = 1$: $[\left\{u^k_{max}(t_1)\right\}, \left\{u^k_{min}(t_1)\right\}] = EMD[x(t_1)]$
for $t_i = 2 : m$ **do** (Online signal from frame 2 to frame m)
$k = 1$ (Starting from the imf_1)
repeat
  repeat
  $$\left.\begin{array}{l}\left\{u^k_{max}(t_{i-1}), z_{max}[x(t_i)]\right\} \\ \\ \left\{u^k_{min}(t_{i-1}), z_{min}[x(t_i)]\right\}\end{array}\right\rangle \rightarrow Hermite[U(t_{i-1}, t_i), L(t_{i-1}, t_i)]$$
$M(t_{i-1}, t_i) = \frac{[U(t_{i-1},t_i)+L(t_{i-1},t_i)]}{2}$
$y(t_i) = x(t_i) - M(t_i)$
  until $y(t_i)$ **Satisfy** (IMF Require)
   $y(t_i) = imf_k, x(t_i) = x(t_i) - y(t_i), k = k + 1$
until $x(t_i)$ **Satisfy** (End Require)
save $\left\{u^k_{max}(t_i)\right\}, \left\{u^k_{min}(t_i)\right\}$

In summary, recursion EMD can be expressed as follows:

$$(imf_i(t_i), u^k_{max}(t_i), u^k_{min}(t_i)) = EMD(x(t_i), u^k_{max}(t_{i-1}), u^k_{min}(t_{i-1})) \quad (1)$$

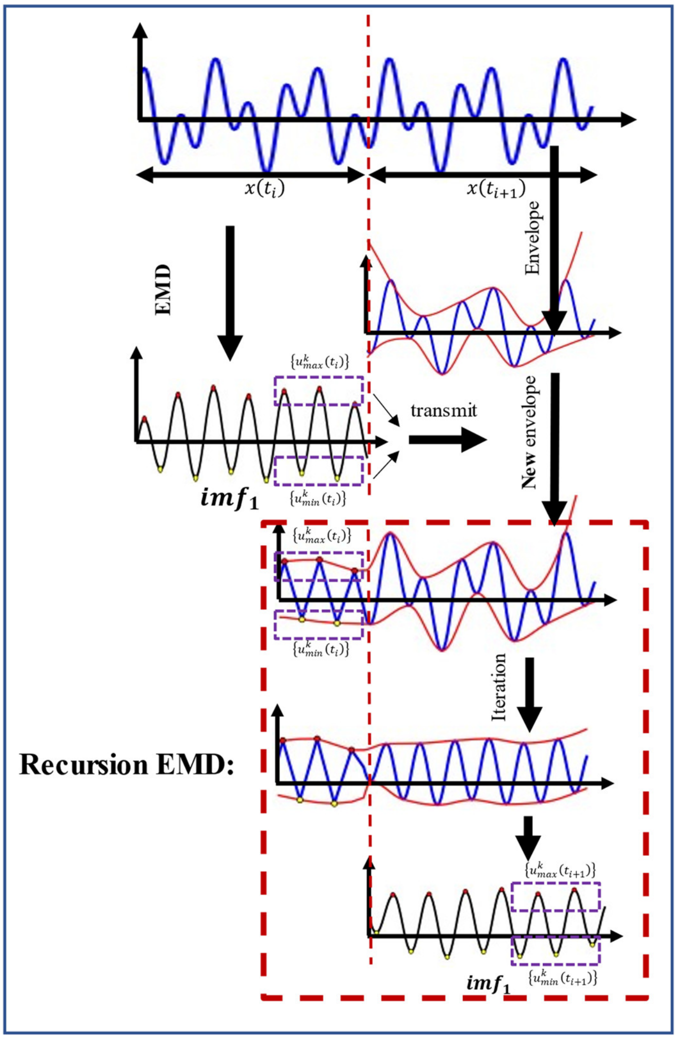


Figure 2. Flow chart of recursion EMD.

2.4. Termination Boundary Effect: Signal Extension Based on ERA Singular Value Truncation

Recursive EMD improves the real-time signal processing and reduces the online computation, but the method only considers the initial boundary effect of the current data frame and does not consider the termination boundary effect; the extreme values with excessive deviation are transmitted to the next frame by recursion, which will affect the subsequent processing and cause errors to accumulate with the increase in calculation steps, affecting the accuracy of the overall decomposition.

The termination boundary effect references the treatment of the EMD boundary effect, using data extension [19]. Extension preferably includes two parts: 1. existing signal noise reduction and sifting; 2. signal extension [27].

To simplify the calculation process, the system in the current calculation frame is considered as a linear system of discrete free vibrations, and the state space equation can be obtained as follows:

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) = \mathbf{A}^{k+1}\mathbf{x}_0, \quad \mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \tag{2}$$

where $\mathbf{x}$ is a n-dimensional state vector and $\mathbf{y}$ is a measurement response vector. $\mathbf{A}$ is the system matrix representing the dynamic characteristics of the system, and $\mathbf{C}$ is the observation matrix transferring the state variable $\mathbf{x}$ to the measured response $\mathbf{y}$. The k represents the kth instant. For the free vibrations, there is no control matrix or force vector. Of note, for each frame data, $\mathbf{A}$, $\mathbf{C}$, $\mathbf{x}_0$ are constant.

Here the eigensystem realization algorithm (ERA) is used to sift the system parameters [19]. However, the purpose of data extension is not to obtain system parameters such as the frequency and damping of the system, but to extend more "appropriate" data. The ERA method is a time-domain modal parameter identification method, which is fast and has strong identification capabilities for the varying frequency signals of each frequency band. The data obtained by this method can retain the frequency, amplitude and other signal characteristics of the original signal as far as possible, which is better compared with the direct mirror extension effect (referred to as ERA extending).

The block Hankel matrix for a one-dimensional signal with a single input is shown:

$$\mathbf{H}(k-1) = \begin{bmatrix} \mathbf{y}(k) & \mathbf{y}(k+1) & \cdots & \mathbf{y}(k+s-1) \\ \mathbf{y}(k+1) & \mathbf{y}(k+2) & \cdots & \mathbf{y}(k+s) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{y}(k+n-1) & \mathbf{y}(k+n-1) & \cdots & \mathbf{y}(k+n+s-2) \end{bmatrix}_{n\times s} \tag{3}$$

The n should be greater than $20f$, s should be as large as possible. When $k = 1$, obtain $\mathbf{H}(0)$:

$$\mathbf{H}(0) = \mathbf{USV}^T \tag{4}$$

Improving signal-to-noise ratio of signals by truncating singular values, where r means the first r singular values. $\mathbf{H}(0)$ can be presented as:

$$\mathbf{H}(0) = \mathbf{USV}^T \approx \mathbf{U}_r\mathbf{S}_r\mathbf{V}_r{}^T \tag{5}$$

Then, the system matrices can be rewritten as:

$$\mathbf{A} = \mathbf{S}_r^{-1/2}\mathbf{U}_r^T\mathbf{H}(1)\mathbf{V}_r^T\mathbf{S}_r^{-1/2}, \quad \mathbf{C} = \mathbf{E}_m^T\mathbf{U}_r\mathbf{S}_r^{1/2}, \quad \mathbf{x}_0 = \mathbf{S}_r^{1/2}\mathbf{V}_r^T\mathbf{E}_1 \tag{6}$$

where $\mathbf{E}_m^T = \begin{bmatrix} \boldsymbol{I}_m & \mathbf{0}_m & \cdots \mathbf{0}_m \end{bmatrix}$, $\mathbf{E}_1^T = \begin{bmatrix} 1 & \mathbf{0} & \cdots \mathbf{0} \end{bmatrix}$, and $\boldsymbol{I}_m$ is unit matrix with the dimension m.

Combined with the state space equation and system parameter $\mathbf{A}, \mathbf{C}, \mathbf{x}_0$, the extension signal $\mathbf{y}(k+t)$ based on the signal characteristics of this computed frame can be obtained.

To prove the advantage of the ERA extension method, three Gaussian noises with average of 0 and standard deviation of 0.5, 1 and 2 are added to the standard signal $x(t) = sin(2\pi 2t) + cos(2\pi 5t)$ to form the simulation signal. The signal is sampled at 100 Hz, and the sampling time is 1 to 5 s. The singular value truncation r of ERA is taken as $0.1n = 5$, and the extension time is 1 s. (a), (c), (e) of Figure 3 show the extension results, and (b), (d), (f) show the extension part details (last 2 s) (Figure 3a–e).

Mean square error (*MSE*) is used to measure the noise reduction effect; the results are shown in Table 1. The method is effective for signals with different noise levels. When the standard deviation of the noise increases, the delaying effect becomes worse because the added noise is close to the amplitude and frequency of the original signal; hence, the original signal is more easily confused with the noise.

Table 1. Error statistics for different signal conditions.

Signal	*MSE*	*MSE* (ERA)	Noise Reduction Rate
Original	0.00	/	/
0.5 standard deviation	≈0.25	≈0.004	99%
1.0 standard deviation	≈1.00	≈0.03 (±0.01)	97%
2.0 standard deviation	≈4.00	≈0.10 (±0.10)	95%

It can be expected that as the variance of the noise increases, the noise signal becomes the main component of the segment signal, and the original signal is annihilated in the noise signal, which is less effective at this time, while the current computational frame can be processed by this method for effective information extension if there is valid information and it occupies the main component.

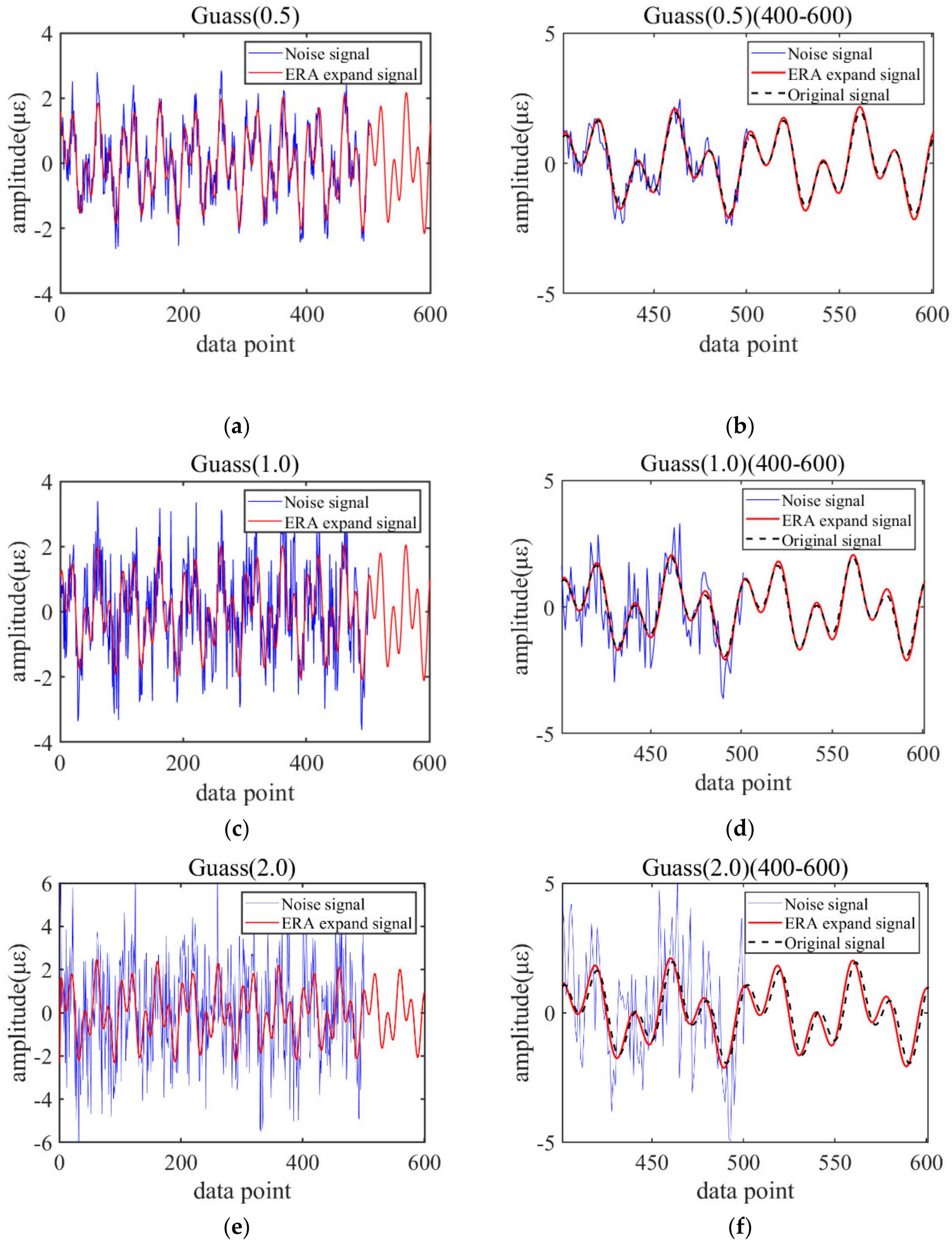


Figure 3. (**a**,**c**,**e**) Signal with standard deviation of 0.5, 1 and 2. (**b**,**d**,**f**) Extension part details.

2.5. Combined Applicability of Recursive EMD and ERA Extension

The recursive EMD optimizes the initial boundary effect of the current frame, while the ERA extension optimizes the termination boundary effect and transmits more accurate extreme information for the recursive EMD of the next frame.

To further verify the effectiveness of the combination of the two methods, a comparison was made with several common methods. The first IMF component of the global EMD and the following types of methods were used for simulation comparison: (1) non-recursive

EMD without extension; (2) non-recursive EMD with mirror extensions; and (3) recursive EMD with ERA extensions. The sifting result is shown in Figure 4a, and the cumulative error compared to the global EMD is shown in Figure 4b.

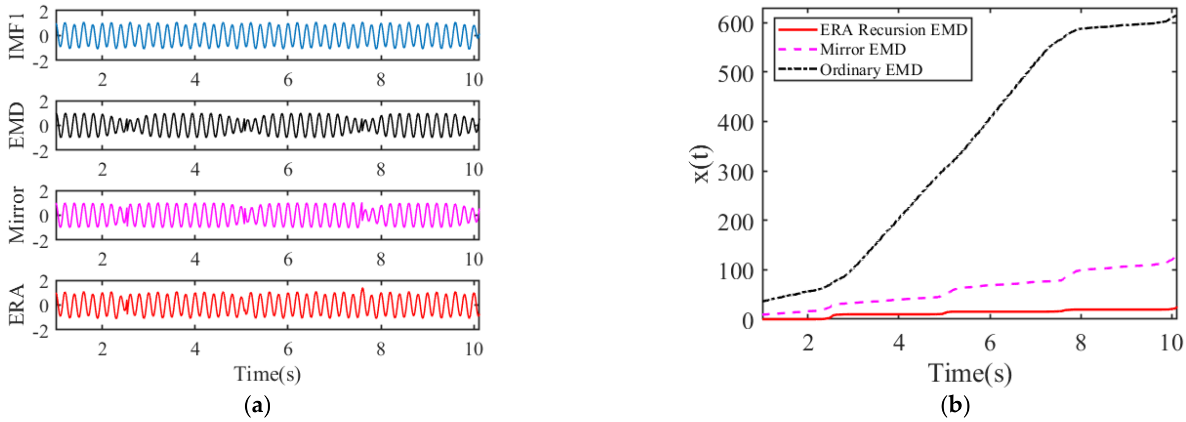


Figure 4. (**a**) The imf-1 results for each method. (**b**) The absolute error accumulation.

The simulation signal $x(t) = sin(2\pi 2t) + cos(2\pi 5t)$ is taken as an example to check the procedure. The signal is sampled at 100 Hz, the sampling time is 1 to 10 s, and the ERA singular value truncation $r = 0.1n = 0.05$ signal length. The above time is also divided into four processing frames to simulate online stream processing; each frame is processed individually and compared with the global EMD batch results.

The mean square errors of the three methods are 0.9123, 0.0287and 0.0001487, and the absolute error accumulation shows that the recursive EMD with ERA extension used in this paper reduces the errors generated by stream processing and has the best effect. Here, only the $1stimf$ are considered, and the trend of the experimental data is the same for others.

3. Online Signal Processing Framework Based on Recursive EMD

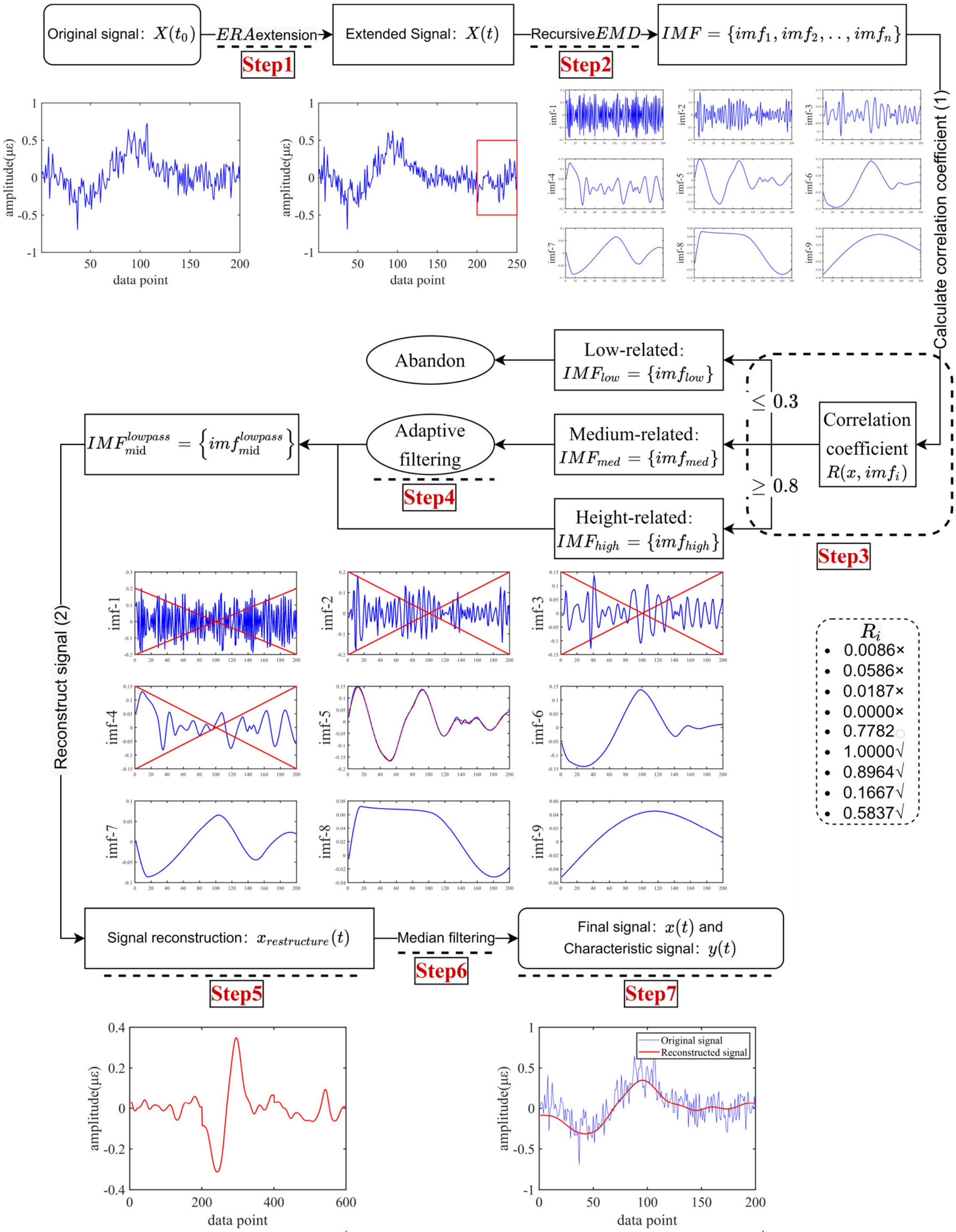
The online recursive EMD signal processing framework (referred to as OREMD) consists of the following seven steps (shown in Figure 5): ERA extending; recursive EMD; selecting effective IMF based on correlation coefficients; adaptive filtering for medium-IMF optimization; signal reconstruction; signal post-processing; and transmission to the next calculation frame.

Step 1:

Extend the current frame signal with ERA to reduce the boundary effect; if the extension signal is extremely fluctuating (usually greater than three times the existing data), it is replaced by mirror extension.

Step 2:

Using recursive EMD, decompose the extended signal, obtaining up to ten IMFs. Ten IMFs are enough for the subsequent process.



Note: (1)$R(x, imf_i) = \left| \frac{\sum_{t=1}^{N}(x(t)-\overline{x})(imf_i(t)-\overline{imf}_i)}{\sqrt{\sum_{t=1}^{N}(x(t)-\overline{x})^2}\sqrt{\sum_{t=1}^{N}(imf_i(t)-\overline{imf}_i)^2}} \right|$

(2): $x_{restructure}(t) = = \sum_{i=1}^{n} imf_{med}^{lowpass} + \sum_{j=m}^{k} imf_{high}$

Figure 5. OREMD diagram. The blue line indicates the original signal, the red line indicates the valid signal after sifting.

Step 3:

The basic idea of the EMD-based sifting method is that the noise signal is contained in the high-frequency IMFs. Removing these IMFs can improve the signal-to-noise ratio and reveal the true value of the signal. There are two categories to obtain the high-frequency IMFs. One is manually specifying the first one or two as the high-frequency IMF(s), which is subjective and only applies to specific signal cases. However, for online signals, each frame of the signal characteristics is different. The other is the cross-correlation coefficient method, which uses the correlation coefficient between the original signal and the IMFs to select high-frequency IMFs [28]. The cross-correlation coefficient is defined as:

$$R(x, imf_i) = \left| \frac{\sum_{t=1}^{N} (x(t) - \overline{x})\left(imf_i(t) - \overline{imf_i}\right)}{\sqrt{\sum_{t=1}^{N} (x(t) - \overline{x})^2}\sqrt{\sum_{t=1}^{N} \left(imf_i(t) - \overline{imf}_i\right)^2}} \right| \tag{7}$$

where $x(t)$ is the original signal, imf_i is the ith IMF, N is the length of $x(t)$, $\overline{x} = \frac{1}{N}\sum_{t=1}^{N} x(t)$ is the average of $x(t)$, and $\overline{imf_i} = \frac{1}{N}\sum_{i=1}^{N} imf_i(t)$ is the average of imf_i.

With the increase in the number of EMD iterations, the correlation between intrinsic modal and noise will also decrease. Conversely, the correlation with the real signal will increase, the correlation coefficient is the result of superposition of two types of correlation, and negatively correlated sexual configuration of information is equally ignored. To further improve the applicability of this method, this paper uses the correlation coefficient to establish evaluation criteria, establishing the foundation for the next processing step, which is as follows:

As the depth of EMD iterations increases, the correlation between the IMFs and the noise decreases; conversely, the correlation with the real signal increases. It is worth noting that the negative correlation is also important, so the absolute value is taken. To further improve the applicability of the method, this paper uses this correlation coefficient to establish the evaluation criteria for the next processing step. All correlation coefficients $R(x, imf_i)$ are normalized to range [0, 1] to obtain $R_0(x, imf_i)$ and classified. The classification results are shown in the following Table 2.

Table 2. Correlation coefficient classification.

Name	R_0	Meaning	Processing
$imf_i(low)$	0.0–0.3	Low-related	Abandon
$imf_i(mid)$	0.3–0.8	Medium-related	Handle
$imf_i(high)$	0.8–1.0	Height-related	Retain

The main significance of this step is to evaluate and determine the valid IMFs.

Step 4:

Medium-related IMFs $imf_i(mid)$ contain some valid information, so it is necessary to optimize medium-related IMFs to keep the process and peak information.

From the perspective of the frequency domain, the larger the $R(x, imf_i)$ the closer the component is to the effective signal, and the frequency domain information distribution condition is closer to the effective signal. Based on this feature, this paper uses adaptive low-pass filtering to optimize the medium-related IMFs $imf_i(mid)$; the specific steps are as follows:

First, record the imf_i with highest $R_0(x, imf_k)$ as the optimal frequency imf_k. When $i < k$, compared to the imf_k, the $imf_i(mid)$ belongs to the high frequency IMFs, which contain too much high frequency information. So, a low-pass filter is used to optimize the $imf_i(mid)$ and improve relevance. The cut-off frequency f_{limit} is the key to the designed low-pass filter, which is stetted by the correlation coefficient $R_0(x, imf_i)$ in this paper

(Algorithm 2). When $i > k$, compared to the imf_k, the $imf_i(mid)$ belongs to the low frequency IMFs, which contain too much low frequency information. So, a high-pass filter can be used to optimize the $imf_i(mid)$. However, the energy decreases with iterative depth increasing, resulting in the effect of optimization not being obvious, so it is reserved directly.

Algorithm 2. Adaptive low-pass filter

Initialize $imf_i, R_0(x, imf_i)_{max} \rightarrow k = i$
if $i < k$ do (low-pass filter)
$S_i(f) = PSD(imf_i)$ (One-sided power spectrum density)
$P_i(k) = \sum_0^k S_i(f), k \in [0 \sim f]$ (Cumulative summation)
$P_{\text{limit}} = R_0(x, imf_i) \times P_i(f)$
find $P_i(k) = P_{\text{limit}} \rightarrow f_{\text{limit}} = k$
$imf_i^{lowpass} = lowpass(imf_i, f_{\text{limit}})$
if $i > k$ do nothing

Low-pass filtering is used to optimize the IMF in terms of energy in the frequency domain. Although the signal amplitude and phase information are not considered, the method is better suited for optimization and has optimized results in most situations.

Step 5:

Obtain $imf_i^{lowpass}$ after the adaptive filtering of $imf_i(mid)$; the signal is sifted as follows:

$$x_{\text{re}}(t) = \sum_{i=1}^{m} imf_i^{lowpass} + \sum_{i=m}^{n} imf_i(high) \tag{8}$$

Step 6:

Sifted signals contain most of the effective information and partial noise, at which time the status of the response signal is specific and the usage requirement, the signal to noise ratio, is further improved by median filtering. This method utilizes the numerical value in the middle value in the data window, which is excellent for the distribution of the outbound point signal, and the specific expression is as follows:

The sifted signal $x_{\text{re}}(t)$ contains most of the valid information and some noise, so the median filtering should be used to further improve the signal-to-noise ratio depending on the specific situation of the signal and the usage requirements. This method uses the median value in the data window, which performs well for outlier signal distributions. The expressions are as follows:

$$X(i) = median(x(i - M/2), \ldots, x(i + M/2)), i = M/2 + 1, M/2 + 2, \ldots, N - M/2 \tag{9}$$

where N is the length of the signal, and M is the length of the window.

Step 7:

Repeat the process of the next calculation frame.

The control parameters of online recursion EMD (OREMD) include the number of singular value truncation r of ERA noise reduction and expansion and the median filter width M of the sifted signal.

When ERA mainly plays the role of expansion, for signals with unknown signal-to-noise ratio, the r of singular values can be taken above $0.8n$ to retain the original information components of the signal as much as possible. For the median filter width M of the sifted signal, a suitable width can be chosen depending on the quality of the sifted signal.

The original signal $x(t)$ is processed by OREMD to obtain $x_{\text{re}}(t) = OREMD(x(t), r, M)$. The framework inherits the high adaptiveness of EMD and requires only a few parameters to meet the requirements for online data.

4. Verification

In this section, standard nonlinear noise test signals are used to verify the applicability of OREMD and to evaluate its effectiveness, which is evaluated by *MSE*. The following changes were made to simulate the online signal processing:

1. Change the standard signal reproduction period to simulate the actual signal time randomness and avoid the influence of periodicity.
2. Compress the standard signal amplitude to simulate the intensity change of the actual signal and avoid the effect of repeatability.
3. Add standard signal noise components to verify the effectiveness of the sifted signal of the framework.
4. Divide the standard signal length and simulate online streaming processing to verify the effect of different time scales.

4.1. Verification of the Proposed Signal Sifting Method

The standard signal is a non-linear non-smooth *Heavy sine* (*Hs*) signal (shown in Figure 6a) with a length of 128, a period of 500, 300, and 200 data points, and three types of amplitude compression: 0.8, 0.6, and 0.4. The Gaussian noise with a mean of 0.5 and a variance of 1 is added to form the test signal *x(t)* (shown in Figure 6b):

$$x(t) = \left\langle \begin{array}{ll} \{Gauss\ noise(0.5, 1)\} & (10a) \\ \{\{0_{500}, 1Hs, 0_{500}, 0.8Hs, 0_{300}, 0.4Hs, 0_{300}, 0.6Hs, 0_{200}, 0.8Hs, 0_{500}\} & (10b) \end{array} \right. \quad (10)$$

where 0_i is a zero value signal with a length of i; $\alpha\ Hs$ indicates standard signal *Heavy sine* (Hs) amplitude compression by a factor of α. Equation (10b) indicates that the signals shown are sequentially articulated to form a new signal. The red line represents the valid *Hs* signal part.

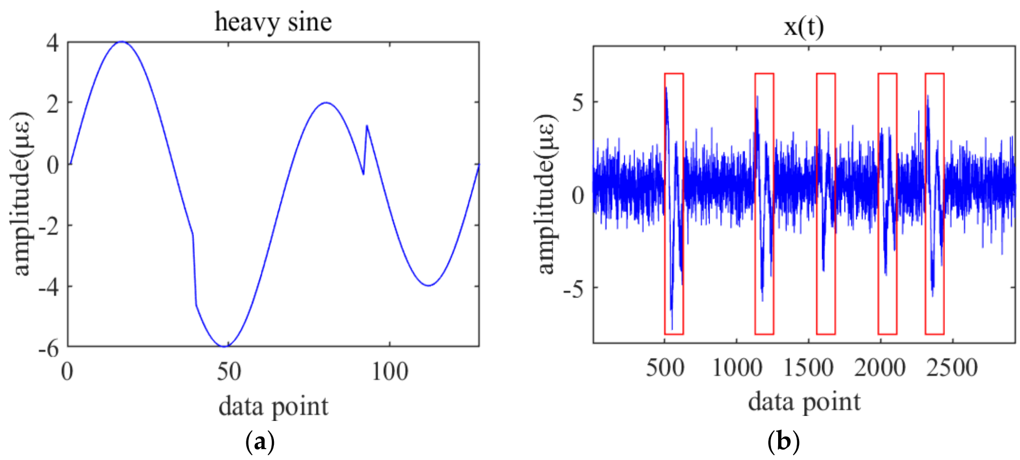


Figure 6. (**a**) Heavy sine signal. (**b**) Test signal $x(t)$.

The signal has a total length of 2940 data points and is divided into five segments $x(t)1-5$ with 588 data points per segment. The parameters of OREMD $n = 200$, $r = 0.9n$, M depends on the effect.

Take the second segment $x(t)-2$ as an example (shown in Figure 7a), which has two ends located in the standard signal change section.

The signal is first extended for 50 data points (shown in Figure 7b), and the extension result extracts the features of the signal as much as possible and continues the waveform characteristics.

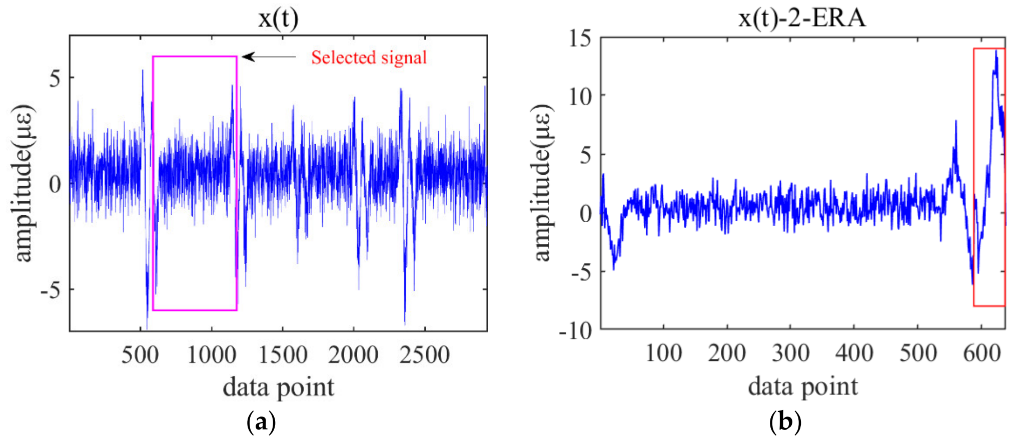


Figure 7. (**a**) Test signal *x(t)*. (**b**) Extended signal; the red window indicates the extension result.

In the next step, recursive EMD decomposition is performed, and the correlation coefficients of IMFs are counted in Table 3.

Table 3. Correlation coefficient of IMFs ($x(t)$-2).

$x(t)$-2	$R(x,imf_i)$	Normalization	$R_{new}(x,imf_i)$	Results	Δ
$imf_1 \sim imf_{10}$	0.0110	0.0000	0.0110	×	0.0000
	0.0266	0.0205	0.0266	×	0.0000
	0.0272	0.0213	0.0272	×	0.0000
	0.7422	0.9607	0.7422	√	0.0000
	0.7721	1.0000	0.7721	√	0.0000
	0.5107	0.6566	0.5107	√	0.0000
	0.5523	0.7112	0.5523	√	0.0000
	0.5585	0.7194	0.5585	√	0.0000
	0.5685	0.7325	0.5685	√	0.0000
	0.5156	0.6630	0.5156	√	0.0000

Results: × (Abandon), √ (Retain).

The maximum correlation coefficient $R(x, imf_i)$ is imf_5, so it is only necessary to optimiz $imf_i(i = 1, 2, 3, 4)$. $R(x, imf_i), (i = 1, 2, 3)$ are less than 0.3 after normalizing, judged as low-related IMFs discarded. $R(x, imf_i), (i = 4)$ is more than 0.8 after normalizing, judged as high-related IMFs retained like the other IMFs.

Since this segment has no filter requirement, another segment $x(t)$-3 is used as an example, and the correlation coefficients of IMFs are counted in Table 4.

Table 4. Correlation coefficient of IMFs ($x(t)$-3).

$x(t)$-3)	$R(x,imf_i)$	Normalization	$R_{new}(x,imf_i)$	Results	Δ
$imf_1 \sim imf_{10}$	0.0040	0.0000	0.0040	×	0.0000
	0.1101	0.1606	0.1101	×	0.0000
	0.0706	0.1009	0.0706	×	0.0000
	0.3571	0.5345	0.4289	○	0.0718
	0.6646	1.0000	0.6646	√	0.0000
	0.5058	0.7596	0.5058	√	0.0000
	0.3417	0.5113	0.3417	√	0.0000
	0.2980	0.4451	0.2980	√	0.0000
	0.2520	0.3754	0.2520	√	0.0000
	/	/	/	/	/

Results: × (Abandon), √ (Retain), ○ (Handle).

The maximum correlation coefficient $R(x, imf_i)$ is imf_5, so it is only necessary to optimiz $imf_i(i = 1, 2, 3, 4)$. $R(x, imf_i), (i = 1, 2, 3)$ are less than 0.3 after normalizing, judged as low-related IMFs discarded. $R(x, imf_i), (i = 4)$ is less than 0.8 after normalizing, judged as medium-related IMFs handled. The $R_{new}(x, imf_i)$ $(i = 4)$ improved 0.0718 after filtering, about 20%, which reflects the effectiveness of the adaptive low-pass filter.

For $x(t) - 2$ the method of OREMD yields $x_{re}(t) = \sum_{i=1}^{m} imf_i^{lowpass} + \sum_{i=m}^{n} imf_i = \sum_{i=5}^{10} imf_i$ (shown in Figure 8a,b). To further compare the sifting effect, the results after the median filter of M = 10 are shown in Figure 8c,d.

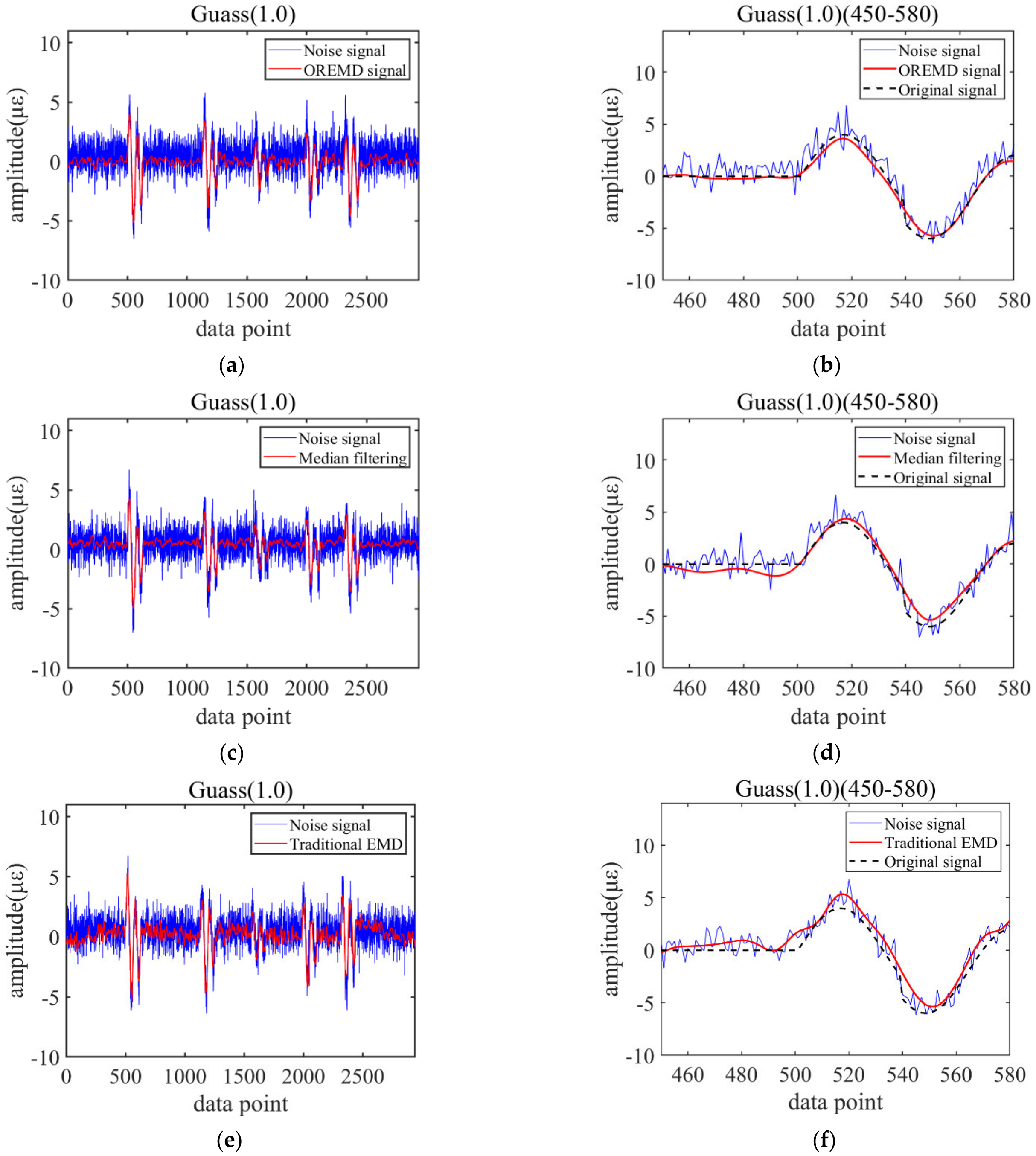


Figure 8. Signal sifting results: (**a**,**b**) OREMD. (**c**,**d**) Median filtering. (**e**,**f**) Traditional EMD.

The MSE of OREMD is 0.2015. The median filter width is taken as M = 18, and the optimal MSE is 0.3183. The MES of traditional EMD is 0.6599. The sieving error using the conventional EMD method is relatively large, so median filtering is used as a comparison in the subsequent content. Intercepting the original effective signal $x(t)[450 - 580]$ segment (Figure 8b,d,f), OREMD sifting is smoother and has higher similarity.

To further verify the sifting effect of different noise levels, Gaussian noise with mean 0.5 and variance 0.5, 1.5, and 2 were added to the original signal to form a new test signal $x_{0.5}(t), x_{1.5}(t), x_2(t)$. The results are shown in Figure 9.

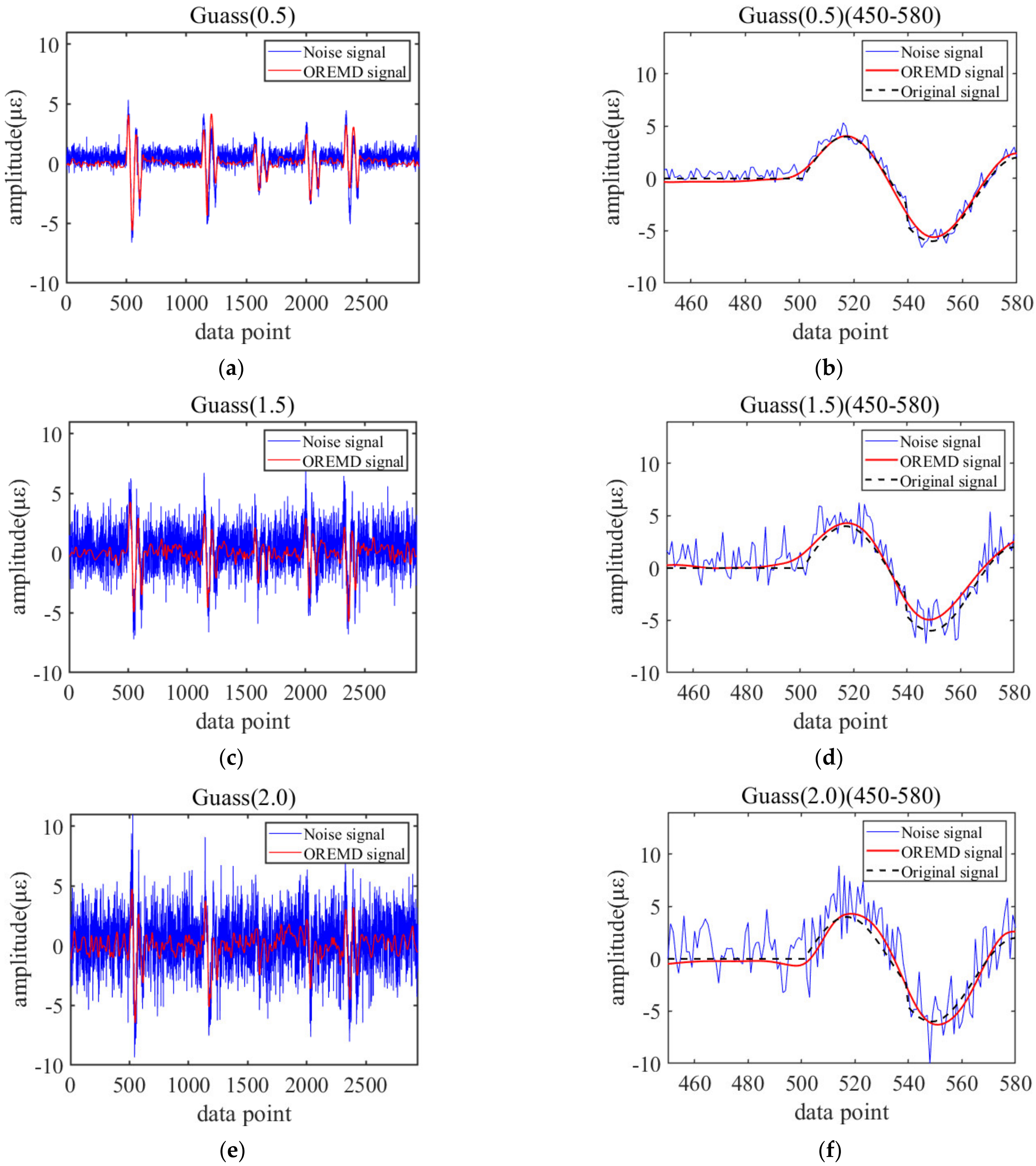


Figure 9. Gaussian noise original signal with different variance and sifted signal: (**a**,**b**) 0.5 standard deviation, (**c**,**d**) 1.0 standard deviation, (**e**,**f**) 2.0 standard deviation.

The *MSE* in the four cases was counted to evaluate the effectiveness of the method and compared with the median filter in Table 5.

Table 5. Sifting effect of different signal sources.

Test Signals	***MSE*-OREMD**	***MSE*-MID**	***MSE*-Original**
Original signal	0.00	0.00	0.00
Gaussian noise with variance 0.5	$r = 0.9n = 90; M = 10$	$M = 14$	0.4944
	0.1776	0.2695	
Gaussian noise with variance 1.0	$r = 0.9n = 90; M = 10$	$M = 16$	1.2327
	0.1666	0.3361	
Gaussian noise with variance 1.5	$r = 0.9n = 90; M = 10$	$M = 20$	2.5724
	0.2426	0.3358	
Gaussian noise with variance 2.0	$r = 0.9n = 90; M = 10$	$M = 22$	4.2093
	0.3234	0.4438	

Due to the random generation of Gaussian noise, the above data are the result of non-extreme processing. The sifting effect by OREMD is significantly better than the direct mean filter sifting. In addition, OREMD retains the local features of the effective signal better; the median filtering has a significant peak-shaving effect. In summary, OREMD is better for signal sifting

4.2. Parameter Sensitivity Analysis

The sensitivity of a parameter is defined as the degree of influence of the preset parameter on the sifting effect. For online systems, the preset parameters are not modified easily during signal processing. Due to the lack of a priori conditions for future information, the pre-defined parameters must be determined at the beginning. When the errors and trends of the data change over time, the variability of the sifting results is obvious if the parameters are too sensitive. Therefore, for large-scale temporal data, parameter sensitivity is one of the decisive factors in determining the quality of signal sifting [10].

The *MSE* is also used to evaluate the quality of the signal sifting, and the main parameters of $x_{\mathrm{re}}(t) = OREMD(x(t), r, M)$ include the truncation value r and the median filter window M. The standard signal is divided into five segments as in the previous section, and the M is taken to be in the range of 2–20 with an interval of 2, for a total of 10; the truncation value r is taken to be in the range of $0.1n \sim n$(10~100) with an interval of $0.1n(10)$, for a total of 10. Considering the randomness of the generated noise error, each case is simulated 10 times, totaling $10 \times 10 \times 10 = 1000$ times, and the results are averaged. The results are shown in Figure 10.

From the results, the larger r and M are, the better the sifting effect; there exists a case of optimal parameters, and the *MSE* of the OREMD is as follows:

$$MSE_{max} = 0.2863,\ MSE_{min} = 0.1678,\ \Delta = 0.1185$$

Median filtered signal:

$$MSE_{max} = 0.3361,\ MSE_{min} = 0.5929, \Delta = 0.2568$$

After comparison, it is shown that the OREMD method in this paper has lower sensitivity with optimal *MSE*.

In addition to the above parameters, the time scale of the computational frame is another important influence on the OREMD. If the time scale is too small, errors are generated by frequent boundary effects, resulting in poor sifting effect, and if the time scale is too large, the time lag affects the real-time processing efficiency.

This section evaluates the effect of signal sifting at different time scales. The standard signal is the same as Formula 14 and divided into five categories of 2, 4, 5, 6 and 10 segments to explore the time scale robustness. *MSE* was used to evaluate the sifting effect, and the results are shown in Table 6.

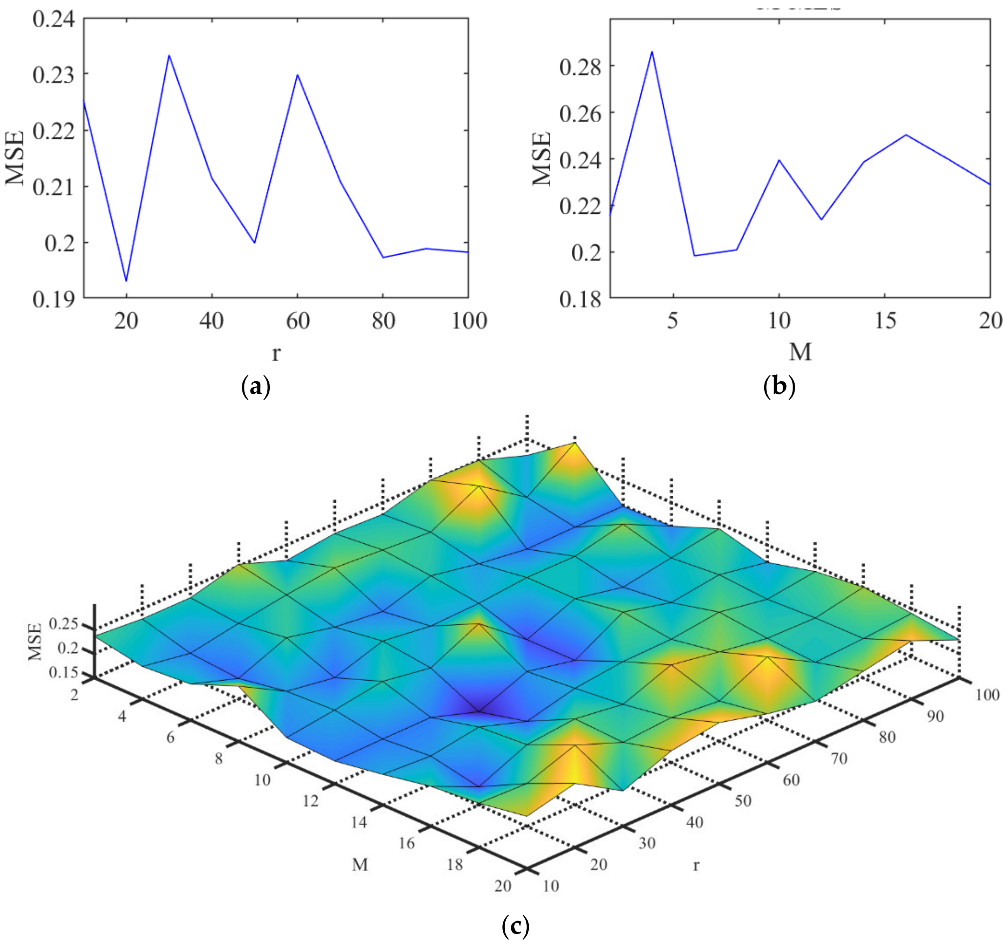


Figure 10. (**a**) M = 6. The trend of *MSE* with r. (**b**) r = 0.9, n = 90. The trend of *MSE* with M. (**c**) Sensitivity chart of OREMD.

Table 6. *MSE* statistics on different time scales.

Time Scale	Data Number	*MSE*-OREMD	*MSE*-MID	*MSE*-Original
1 segment	2940	0.1274	0.3614	1.2620
2 segments	1470	0.1357	0.3459	1.2355
4 segments	735	0.1508	0.3453	1.2442
5 segments	588	0.2223	0.3470	1.2405
6 segments	490	0.1987	0.3658	1.2666
10 segments	294	0.2443	0.3390	1.2510

Note: The OREMD (r = 0.9n, M = 10).

From the results, the original signal of global OREMD works best, and the *MSE* tends to rise with more segments of the signal decomposition, but there exist individual time scales that can be optimal between the interval and error balance.

According to the calculation principle, the *MSE* increases as the time scale decreases, but here the *MSE* of six segments is smaller than the seven segments, which is mainly caused by the error of the demarcation point after analysis, and the specific division effect is shown in Figure 11.

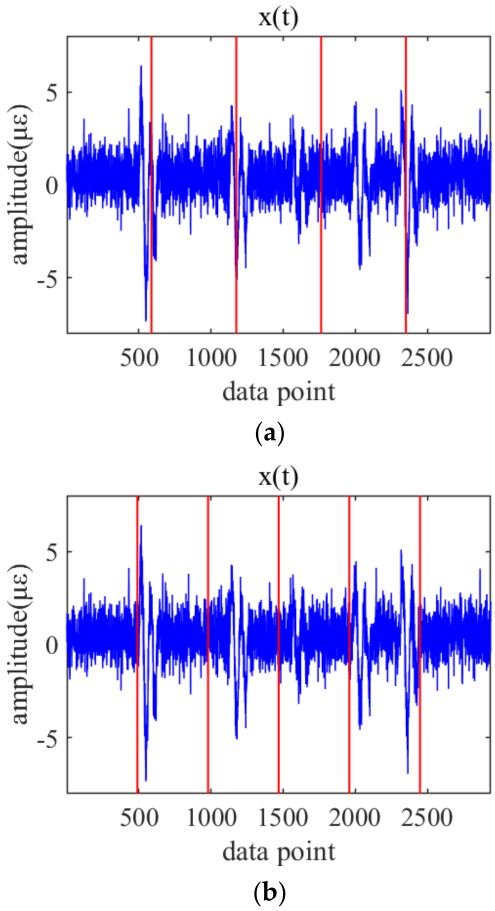


Figure 11. Signal division: (**a**) five segments (**b**) six segments.

When the original signal is divided into five segments, the first, second, and fourth demarcation points are in the segments of the standard signal, where the signal characteristics are more volatile compared to the noisy signal and the amplitude changes significantly, resulting in generating more error and affecting the quality of the sifted signal. When the signal is divided into six segments, the demarcation point is in the smooth segment of the signal, resulting in error being smaller. Through the above analysis, the recommended time scale is about three to four times that of the effective signal, at which time the calculation efficiency is in the high range, and the overall information sifting is better.

After the above theoretical data validation, the validity, parameter sensitivity, and time-scale robustness of OREMD were evaluated. The results showed a significant effect on both data and visual quality.

5. Application in Strain Monitoring Data Processing of Fabricated Girder Bridge

5.1. Project Background

Transverse collaborative working performance is one of the important indicators of the service health of assembled girder bridges, and the transverse collaborative working performance can be judged by using the load condition combined with the structural characteristics of the bridge itself [29]. The actual traffic loads are obtained through the bridge health monitoring system; however, some of the side span sensors are affected by noise and other effects, resulting in poor data quality. Therefore, the above OREMD is applied to the real-time data from the strain sensors of Wuhan Daishan Second Bridge to screen out the effective traffic strain signals for subsequent evaluation.

In this section, the OREMD is applied to the real-time data from the strain sensors of the Wuhan Daishan Second Bridge, which is an assembled reinforced steel three-span continuous T-beam bridge with a 33.1 m side span and 40.13 m main span. The sensors are arranged at the bottom of each T-beam in the side span, which is about 50 cm from the bottom of the beam to protect the cables (Figure 12). The sampling frequency is 20 Hz and the data are from 19 November 2020.

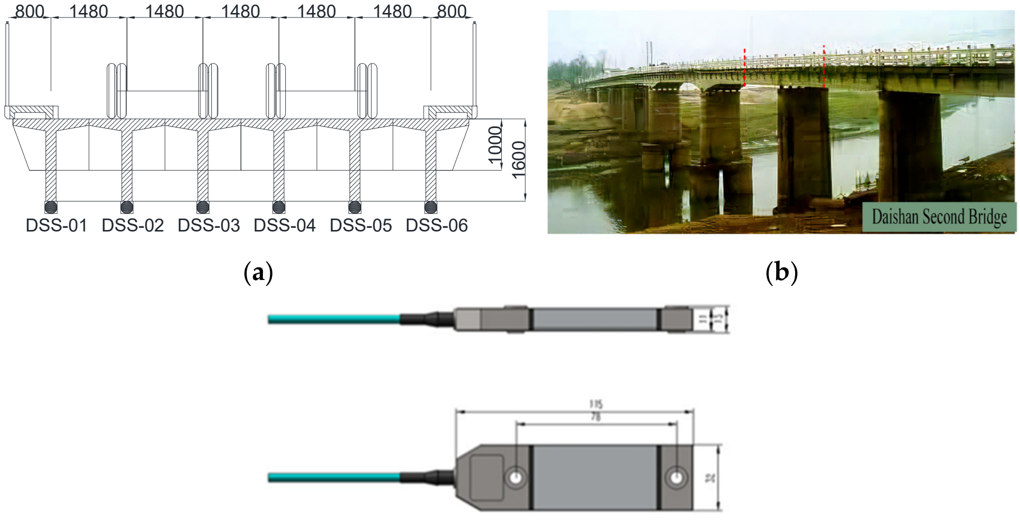
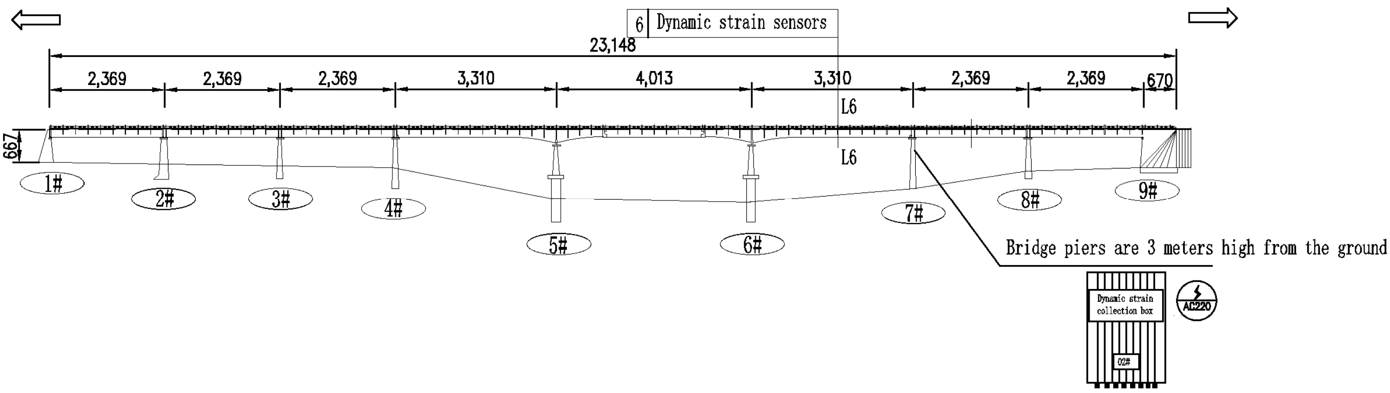


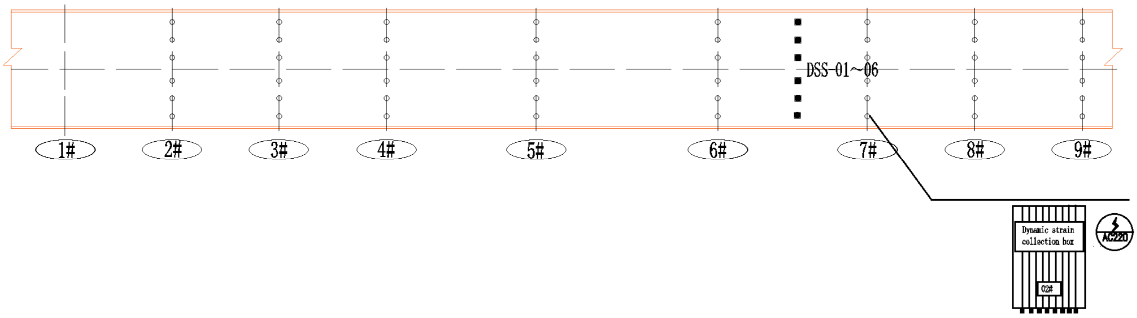
Figure 12. (**a**,**b**) Diagram of the sensor arrangement; (**c**) resistive surface strain gauge.

The strain sensor system for the bridge adopts resistive surface strain gauges, and the sensors are installed at the bottom of the girder by surface-mounting. The concrete surface is polished and de-dusted before installation to ensure the stability of the sensor. During the operation of the sensor, the vehicle passes over the cross section of the sensor arrangement on the bridge deck, and the sensor records the structural response to generate an electrical signal, which is converted into a dynamic strain signal through the on-site strain-gathering box around the clock and then transferred to the workstation computer database for extraction and analysis in real time through the cable (Figure 13).



(**a**)

Figure 13. *Cont.*



(**b**)

Figure 13. (**a**) Elevation layout of structural stress monitoring measurement points. (**b**) Plan layout of structural stress monitoring measurement points.

5.2. Traffic Induced Strain Component Sifting

The data come from half-bridge sensors DSS-01, DSS-02, and DSS-03. DSS-01 is located at the far side, where a certain distance exists from the main lanes, and the signal-to-noise ratio of these data are low because the traffic load signal amplitude is small and the system itself is noisy.

To achieve clearer calculation results, DSS-01 was processed to intercept the data from 00:01:40 to 00:10:00 of the day, and the traffic signal characteristics existed within the signal range of this segment. This signal was divided into 20 segments DSS-01-1~DSS-01-20 for OREMD processing, with an average of 500 data points per segment and a duration of 25 s. The raw data are shown in Figure 14a.

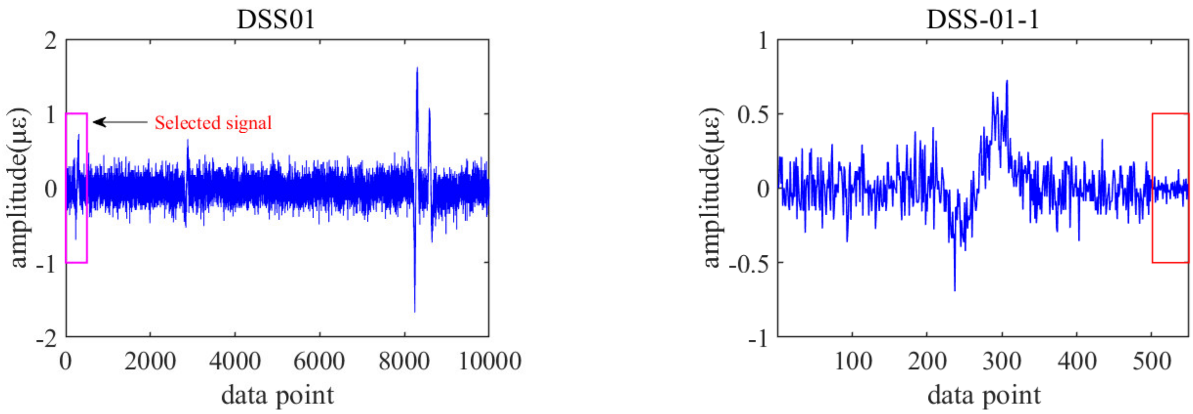


Figure 14. (**a**) Original signal; the purple window shows the signal amplification. (**b**) Signal amplification area; the red window shows the extension result.

Using DSS-01-1 as an example (this part is enlarged from the purple window in Figure 14a to obtain Figure 14b), first take the last 200 data points of the current data for ERA extension (shown in Figure 14b red window), the $hankle(n \times s)$ matrix $n = 80$, $s = 120$. Take $r = 0.5n = 40$ and extend 50 data points backward. The extended data points are in line with the characteristics of the noise signal fluctuating around amplitude 0 with certain accuracy.

In the next step, recursive EMD decomposition is performed, and the correlation coefficients of IMFs $R_{new}(x, imf_i)$ are counted in Table 7.

Table 7. The correlation coefficient of DSS-01-1.

DSS-01-1	$R(x,imf_i)$	Normalization	$R_{new}(x,imf_i)$	Results	Δ
$imf_1 \sim imf_{10}$	0.0293	0.0000	0.0293	×	0.0000
	0.1311	0.1135	0.1311	×	0.0000
	0.3393	0.3457	0.3732	○	0.0340
	0.9259	1.0000	0.9259	√	0.0000
	0.7888	0.8471	0.7888	√	0.0000
	0.5518	0.5828	0.5518	√	0.0000
	0.3444	0.3514	0.3444	√	0.0000
	0.2078	0.1991	0.2078	√	0.0000
	/	/	/	/	/
	/	/	/	/	/

Results: × (Abandon), √ (Retain), ○ (Handle).

The maximum correlation coefficient $R(x, imf_i)$ is imf_4, so it is only necessary to optimize $imf_i(i = 1, 2, 3)$. $R(x, imf_i), (i = 1, 2)$ are less than 0.3 after normalizing, judged as low-related IMFs discarded. $R(x, imf_i), (i = 3)$ is less than 0.8 after normalizing, judged as medium-related IMFs handled. The $R_{new}(x, imf_i)(i = 3)$ improved 0.0340 after filtering, about 10%, reflecting the effectiveness of the adaptive low-pass filter.

Sifting yields $x_{\mathrm{re}}(t) = imf_3^{lowpass} + \sum_{i=4}^{8} imf_i$. The result is shown in Figure 15a.

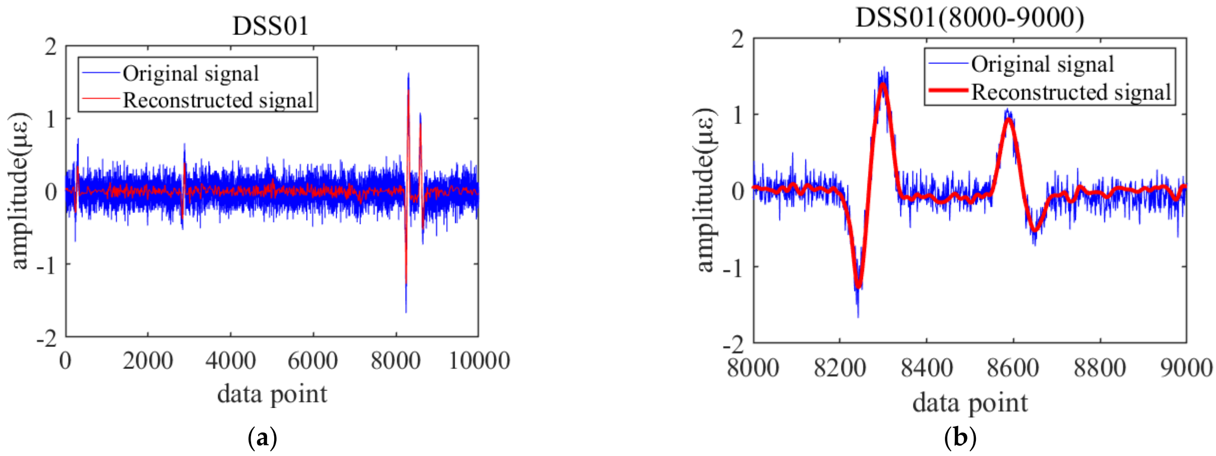


Figure 15. Signal sifting: (**a**) *DSS*01, (**b**) *DSS*01 (8000-9000).

The 8000-9000 fragment of DSS-01-1 is intercepted as Figure 15b. The traffic load signal is smooth without obvious noise interference in this fragment, and without median filtering processing, the peak feature information of the effective signal is better. The results show that the OREMD sifting yields accurate traffic load information.

DSS-02 and DSS-03 are located near the main lane in the cross-bridge direction, so traffic signal amplitude is greater compared to noise, which means the high signal-to-noise ratio. The sifting results are shown as Figure 16.

Due to the high signal-to-noise ratio of the original signal, the sifted signal is accurate and more reliable, and the peak weakening is smaller.

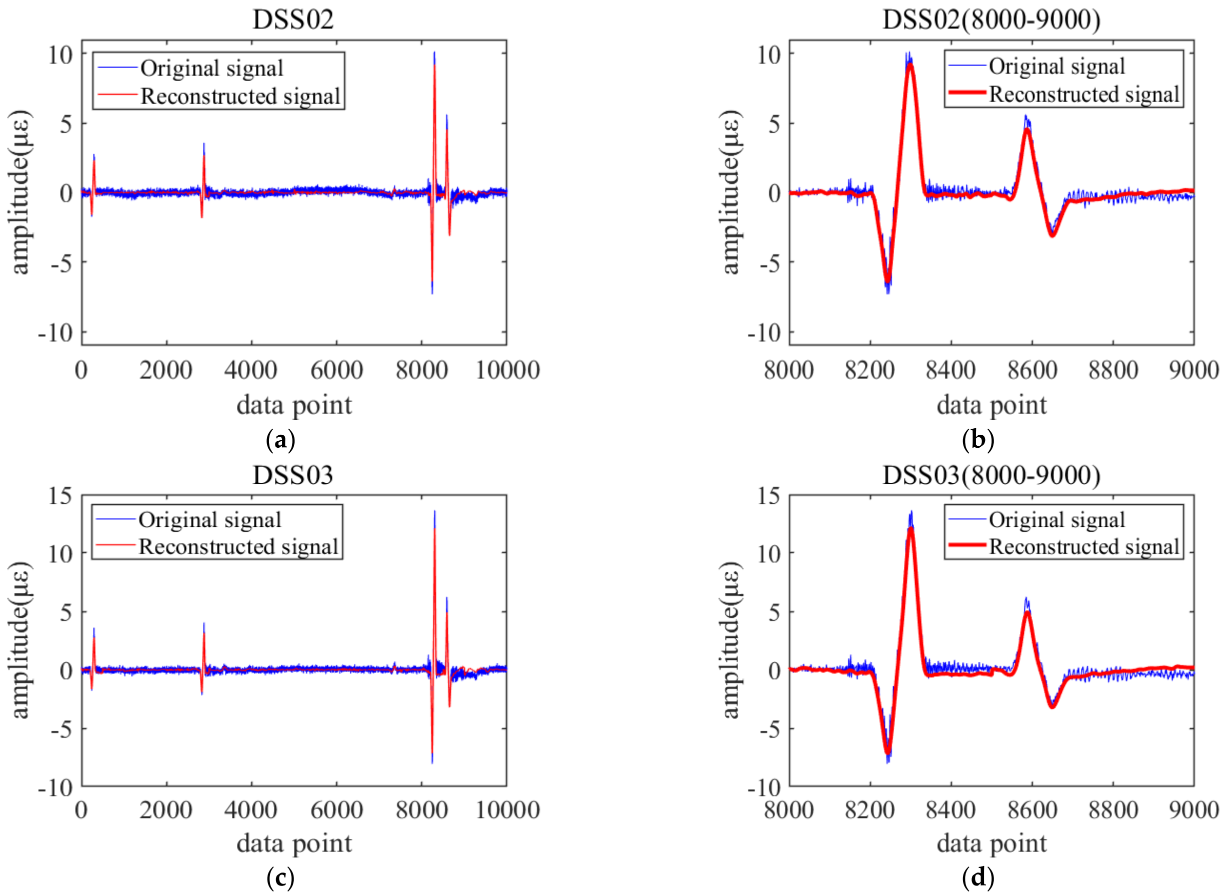


Figure 16. Signal sifting: (**a**,**b**) *DSS*02, (**c**,**d**) *DSS*03.

5.3. Transverse Collaborative Working Performance

By analyzing the correlation between the strains at the bottom of each beam under traffic load, a simple judgment can be made on the performance of transverse co-working, and direct correlation analysis of *DSS*01, *DSS*02, and *DSS*03 above is obtained as follows:

$$\rho_{ij} = \begin{bmatrix} 1.000 & 0.663 & 0.655 \\ & 1 & 0.961 \\ & & 1 \end{bmatrix} \overset{\text{OREMD}}{\rightarrow} \rho_{ij} = \begin{array}{cc} DSS01\,DSS02\,DSS03 & \\ \begin{bmatrix} 1.000 & 0.928 & 0.909 \\ & 1 & 0.977 \\ & & 1 \end{bmatrix} & \begin{array}{c} DSS01 \\ DSS02 \\ DSS03 \end{array} \end{array} \tag{11}$$

The sifted results show that the traffic signals are more correlated. The correlation between *DSS*01 and *DSS*02 is stronger than the correlations with *DSS*03, which is in line with the actual situation.

In addition, the trend of each data correlation can be monitored in real-time through long-term data monitoring, which can be used to evaluate the horizontal synergistic performance of each piece of the main beam in the operation state.

6. Conclusions and Outlook

The method proposed in this paper is not only applicable to bridge health monitoring systems, but also to other building types. During the service period of some building structures, the strain effect waveforms caused by live loads are often significantly different from those caused by other loads, so we can use these time domain waveform differences to screen the strain components under different loads using the method proposed in this paper. For example, for a row-frame industrial plant structure with crane beams, the strain waveforms generated by the crane beams during the travel of the crane beams are significantly different from those generated by other causes, and we can use the method in this paper to screen the strain components generated by the working load of the crane beams online in real time. Another example comes from the fact that large-span arena structures, especially stadium-type structures, are often exposed to strain effects caused by crowd activity, especially when the crowd generates rhythmic movements, and the strain effect waveforms have distinctive features that distinguish them from other loads. The screening of this effect component is performed using the method suggested in this paper.

In this paper, we propose a real-time signal sifting framework OREMD for an online monitoring environment, which retains the adaptive characteristics of EMD and is suitable for signal sifting of single load effect components in an online monitoring environment. The main findings are as follows:

1. The recursive EMD method and the ERA expansion method are used to effectively solve the docking difficulties between the decomposition components in the online monitoring environment and reduce the errors caused by boundary effects.
2. Improving the utilization of IMF by the adaptive filtering method of the IMF correlation coefficient ensures the quality of signal reconstruction in online monitoring environments and improves the automation of signal processing.
3. Numerical simulation cases show that the method is highly adaptable to signals with different noise levels, different reproduction periods, and different intensities, and is robust to signals of different lengths on time scales. For strain monitoring signals of engineering structures, the OREMD method has few pre-set parameters and is highly adaptive for engineering implementation. Finally, for other load effect signals, the effectiveness of the proposed method in this paper needs further validation.

Author Contributions: Conceptualization, D.D. and C.W.; formal analysis, C.W.; investigation, C.W.; resources, D.D., Y.C. and R.P.; data curation, D.D.; writing—original draft preparation, C.W.; writing— review and editing, C.W. and D.D.; visualization, C.W.; supervision, D.D. and R.P.; project administration, D.D.; funding acquisition, D.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the technical consulting project of China Railway Fourth Institute Survey and Design Group Co., Ltd. "Smart urban rail bridge health monitoring and operations management platform" (2020K172) and central college basic research fund (20210205) and Shanghai Urban Construction Design and Research Institute (Group) Co., Ltd. Project "Feasibility Study of High Precision Traffic Flow Monitoring System for Bridges" and China Railway Engineering Design and Consulting Group Co., Ltd. Project "Research and Development of bridge management system based on perceptual neural network".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kaya, Y.; Safak, E. Real-time analysis and interpretation of continuous data from structural health monitoring (SHM) systems. *Bull. Earthq. Eng.* **2015**, *13*, 917–934. (In English) [CrossRef]
2. Proakis, J.G.; Manolakis, D.G. *Digital Signal Processing, 4e: Principles, Algorithms, and Applications*; Pearson Education: London, UK, 1996.
3. Boudraa, A.-O.; Cexus, J.-C. EMD-Based Signal Filtering. *IEEE Trans. Instrum. Meas.* **2007**, *56*, 2196–2202. [CrossRef]
4. Yu, X.; Dan, D. Block-wise recursive APES aided with frequency-squeezing postprocessing and the application in online analysis of vibration monitoring signals. *Mech. Syst. Signal Process.* **2022**, *162*, 108063. [CrossRef]
5. Abrecht, D.G.; Schwantes, J.M.; Kukkadapu, R.K.; McDonald, B.S.; Eiden, G.C.; Sweet, L.E. Real-time noise reduction for Mössbauer spectroscopy through online implementation of a modified Kalman filter. *Nucl. Instrum. Methods Phys. Res. Sect. A Accel. Spectrometers Detect. Assoc. Equip.* **2015**, *773*, 66–71. [CrossRef]
6. Braun, S.; Habets, E.A. Linear prediction-based online dereverberation and noise reduction using alternating Kalman filters. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2018**, *26*, 1119–1129. [CrossRef]

7. Nounou, M.N.; Bakshi, B.R. On-line multiscale filtering of random and gross errors without process models. *Aiche J.* **1999**, *45*, 1041–1058. (In English) [CrossRef]
8. Paleologu, C.; Benesty, J.; Ciochină, S. Adaptive filtering for the identification of bilinear forms. *Digit. Signal Process.* **2018**, *75*, 153–167. [CrossRef]
9. Liu, C.; Zhang, Z.; Tang, X. Sign normalised spline adaptive filtering algorithms against impulsive noise. *Signal Process.* **2018**, *148*, 234–240. [CrossRef]
10. Ma, X.; Hu, J.; Zhang, L. EMD-based online Filtering of Process Data. *Control Eng. Pract.* **2017**, *62*, 79–91. [CrossRef]
11. Duan, W.-Y.; Huang, L.-M.; Han, Y.; Huang, D.-T. A hybrid EMD-AR model for nonlinear and non-stationary wave forecasting. *J. Zhejiang Univ.–SCIENCE A* **2016**, *17*, 115–129. [CrossRef]
12. Al-Musaylh, M.S.; Deo, R.C.; Li, Y.; Adamowski, J.F. Two-phase particle swarm optimized-support vector regression hybrid model integrated with improved empirical mode decomposition with adaptive noise for multiple-horizon electricity demand forecasting. *Appl. Energy* **2018**, *217*, 422–439. [CrossRef]
13. Huang, N.E.; Shen, Z.; Long, S.R.; Wu, M.C.; Shih, H.H.; Zheng, Q.; Yen, N.-C.; Tung, C.C.; Liu, H.H. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proc. R. Soc. A—Math. Phys. Eng. Sci.* **1998**, *454*, 903–995. (In English) [CrossRef]
14. Boudraa, A.O.; Cexus, J.C.; Saidi, Z. EMD-Based Signal Noise Reduction. *Proc. World Acad. Sci. Eng. Technol.* **2013**, *1*, 93.
15. Shrivastava, Y.; Singh, B. A comparative study of EMD and EEMD approaches for identifying chatter frequency in CNC turning. *Eur. J. Mech.—A/Solids* **2019**, *73*, 381–393. [CrossRef]
16. Boda, S.; Mahadevappa, M.; Dutta, K. A hybrid method for removal of power line interference and baseline wander in ECG signals using EMD and EWT. *Biomed. Signal Process. Control Eng. Pract.* **2021**, *67*, 102466. [CrossRef]
17. Jiang, Z.; Che, J.; Wang, L. Ultra-short-term wind speed forecasting based on EMD-VAR model and spatial correlation. *Energy Convers. Manag.* **2021**, *250*, 114919. [CrossRef]
18. Wang, J.; Du, G.; Zhu, Z.; Shen, C.; He, Q. Fault diagnosis of rotating machines based on the EMD manifold. *Mech. Syst. Signal Process.* **2020**, *135*, 106443.
19. Huang, D.J.; Zhao, J.; Su, J.L. Practical implementation of Hilbert-Huang Transform algorithm. *Acta Oceanol. Sin.* **2003**, *22*, 1–14.
20. Han, J.; Zheng, P.; Wang, H. Structural modal parameter identification and damage diagnosis based on Hilbert-Huang transform. *Earthq. Eng. Eng. Vib.* **2014**, *13*, 101–111. [CrossRef]
21. Bican, B.; Yaslan, Y. A hybrid method for time series prediction using EMD and SVR. In Proceedings of the 2014 6th International Symposium on Communications, Control and Signal Processing (ISCCSP), Athens, Greece, 21–23 May 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 566–569.
22. Duan, W.-Y.; Huang, L.-M.; Han, Y.; Zhang, Y.-H.; Huang, S. A hybrid AR-EMD-SVR model for the short-term prediction of nonlinear and non-stationary ship motion. *J. Zhejiang Univ.-Sci. A* **2015**, *16*, 562–576. [CrossRef]
23. Huang, J.; Xiao, Q.; Liu, J.; Wang, H. Modeling heat transfer properties in an ORC direct contact evaporator using RBF neural network combined with EMD. *Energy* **2019**, *173*, 306–316.
24. Nie, Z.; Shen, F.; Xu, D.; Li, Q. An EMD-SVR model for short-term prediction of ship motion using mirror symmetry and SVR algorithms to eliminate EMD boundary effect. *Ocean. Eng.* **2020**, *217*, 107927.
25. Luo, X.; Gan, W.; Wang, L.; Chen, Y.; Meng, X. A Prediction model of structural settlement based on EMD-SVR-WNN. *Adv. Civ. Eng.* **2020**, *2020*, 8831965. [CrossRef]
26. Fan, G.-F.; Peng, L.-L.; Hong, W.-C.; Sun, F. Electric load forecasting by the SVR model with differential empirical mode decomposition and auto regression. *Neurocomputing* **2016**, *173*, 958–970. [CrossRef]
27. Jin, M.; Chen, W.; Brake, M.R.W.; Song, H. Identification of Instantaneous Frequency and Damping from Transient Decay Data. *J. Vib. Acoust.* **2020**, *142*, 51111.
28. Jia, R.S.; Zhao, T.B.; Sun, H.M.; Yan, X.H. Micro-seismic signal denoising method based on empirical mode decomposition and independent component analysis. *Chin. J. Geophys.-Chin. Ed.* **2015**, *58*, 1013–1023. (In Chinese)
29. Dan, D.H.; Xu, Z.Y.; Zhang, K.L.; Yan, X.F. Monitoring Index of Transverse Collaborative Working Performance of Assembled Beam Bridges Based on Transverse Modal Shape. *Int. J. Struct. Stab. Dyn.* **2019**, *19*, 1950086. (In English) [CrossRef]