

A Class of Algorithms for Continuous Wavelet Transform Based on the Circulant Matrix

Hua Yi ¹ , Shi-You Xin ² and Jun-Feng Yin ^{2,*}

¹ Department of Mathematics, Jinggangshan University, Ji'an 343009, Jiangxi Province, China; yihua@whu.edu.cn

² Department of Mathematics, Tongji University, Shanghai 200092, China; 1334065@tongji.edu.cn

* Correspondence: yinjf@tongji.edu.cn

Received: 20 November 2017; Accepted: 22 February 2018; Published: 27 February 2018

Abstract: The Continuous Wavelet Transform (CWT) is an important mathematical tool in signal processing, which is a linear time-invariant operator with causality and stability for a fixed scale and real-life application. A novel and simple proof of the FFT-based fast method of linear convolution is presented by exploiting the structures of circulant matrix. After introducing Equivalent Condition of Time-domain and Frequency-domain Algorithms of CWT, a class of algorithms for continuous wavelet transform are proposed and analyzed in this paper, which can cover the algorithms in JLAB and WaveLab, as well as the other existing methods such as the *cwt* function in the toolbox of MATLAB. In this framework, two theoretical issues for the computation of CWT are analyzed. Firstly, edge effect is easily handled by using Equivalent Condition of Time-domain and Frequency-domain Algorithms of CWT and higher precision is expected. Secondly, due to the fact that linear convolution expands the support of the signal, which parts of the linear convolution are just the coefficients of CWT is analyzed by exploring the relationship of the filters of Frequency-domain and Time-domain algorithms, and some generalizations are given. Numerical experiments are presented to further demonstrate our analyses.

Keywords: circulant matrix; continuous wavelet transform; linear convolution; circular convolution; FFT

1. Introduction

In recent years, different Time-frequency representations, such as, empirical mode decomposition [1], wavelet transform [2] and its variants, empirical wavelet transform [3,4], synchrosqueezed wavelet transforms [5], have been used for analyzing nonlinear and non-stationary signals. To name only a few, the method of fused empirical mode decomposition and wavelets is applied to detection-location of damage in a truss-type structure [6]; Wavelet transform is used for the pattern recognition for diagnosis, condition monitoring and fault detection [7–9]. It is also used to design an algorithm for Brain-computer interfacing [10], to detect the exact onset of chipping of the cutting tool from the workpiece profile [11], to determine the length of piles [12]; Synchrosqueezed wavelet transform is used for global and local health condition assessment of structures [13], for modal parameters identification of smart civil structures [14].

This paper gives priority to the computation of continuous wavelet transform (CWT) for Morlet-type wavelets. However, for this type of wavelets, it is not possible to use a multiresolution framework for the computation of CWT [15]. Michael Unser [16] used Exponentials or B-spline window to approximate Gabor window, which can achieve $\mathcal{O}(N)$ complexity per scale. Another kind of method for the computation of CWT with Morlet-type wavelets concerns directly discretizing the

integral expression of Wavelet transform. These methods allow us to use arbitrary values for the scale variable, and require the explicit expression of wavelet function and cannot deal with those wavelets without analytic expressions such as Daubechies wavelets [17]. This kind of method include Time-domain methods [17] and Frequency-domain methods [18].

Frequency-domain methods for CWT are also widely used in the freewares such as JLAB (available online <http://www.jmlilly.net>), Wavelab (available online [http://www-stat.stanford.edu/~sim\\$wavelab](http://www-stat.stanford.edu/~sim$wavelab)), which can achieve satisfactory precision while the edge effect, defined in [18], may occur and the complexity of those methods is $\mathcal{O}(N \log(N))$ where N is the length of the signal.

The function *cwt* in wavelet toolbox of MATLAB, which has poorer precision than Frequency-domain methods mentioned above, is a representation of the Time-domain method [17]. For the details of the comparison of the different methods, we refer to [17].

The computation of linear convolution can be realized by using the circular convolution, while the computation of circular convolution can be computed by using the FFT-based fast method [19]. In this paper, a novel and simple method is given to prove the FFT-based fast method of linear convolution by exploiting the structures of circulant matrix. After introducing Equivalent Condition of Time-domain and Frequency-domain Algorithms of CWT, a class of algorithms for continuous wavelet transform are proposed and analyzed, which can cover the algorithms in JLAB and WaveLab, as well as the other existing methods such as the *cwt* function in the toolbox of MATLAB. In this framework, two theoretical issues for the computation of CWT are analyzed.

Firstly, by using Equivalent Condition of Time-domain and Frequency-domain Algorithms of CWT to design Frequency-domain Algorithm, the edge effect, defined in [18], can be avoided. To be specific, in [18], the time series is padded with sufficient zeroes to bring the total length N up to the next-higher power of two, thus limiting the edge effects and speeding up the Fourier transform. In fact, the same number of zeros is padded for all scales in [18], which may cause some troubles, for example, increasing the amount of data to process, and consequently the computational complexity. In this article, the time series (the data) is padded with $2aT$ zeros while a is the scale. These two zero padding methods are compared in Remark 2 of this paper.

Secondly, due to the fact that linear convolution expands the support of the signal, which parts of the linear convolution are just the coefficients of CWT is analyzed by exploring the relationship of the filters of Frequency-domain and Time-domain algorithms (see Theorem 4), and some generalizations are given (see Theorem 5).

This paper is organized as follows. Section 2 gives some definitions and theorems concerning circulant matrix and linear convolution. Section 3 analyzes algorithms of continuous wavelet transform. Section 4 presents numerical experiments to demonstrate our results and finally, we end this paper with conclusions and discussions in Section 5.

2. Primary Definitions and Theorems

Definition 1. Circulant matrix: An $N \times N$ circulant matrix \mathbf{C} takes the form [20,21]

$$\mathbf{C} = \begin{bmatrix} c_0 & c_{N-1} & \cdots & c_2 & c_1 \\ c_1 & c_0 & c_{N-1} & & c_2 \\ \vdots & c_1 & c_0 & \ddots & \vdots \\ c_{N-2} & & \ddots & \ddots & c_{N-1} \\ c_{N-1} & c_{N-2} & \cdots & c_1 & c_0 \end{bmatrix} \quad (1)$$

A circulant matrix is fully specified by one vector, c , which appears as the first column of \mathbf{C} .

Definition 2. Discrete Fourier transform(DFT): The sequence of N complex numbers $[x_0, x_1, \dots, x_{N-1}]^T$ is transformed into an N -periodic sequence of complex numbers [22]

$$X_k = \sum_{n=0}^{N-1} x_n \omega_k^{-n}, k = 0, 1, \dots, N-1, \quad (2)$$

where $i = \sqrt{-1}$, $\omega_k = e^{\frac{2\pi ki}{N}}$ is the N -th root of unity.

Equation (2) can be written as

$$\begin{pmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{pmatrix} = \mathbf{F}_N \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{pmatrix} \quad (3)$$

where \mathbf{F}_N is a $N \times N$ matrix, defined as

$$\mathbf{F}_N(k, j) = \omega_{k-1}^{-(j-1)}, \quad k, j = 1, 2, \dots, N. \quad (4)$$

It is easy to verify that

$$\mathbf{F}_N^{-1} = \frac{1}{N} \mathbf{F}_N^* \quad (5)$$

where “*” means conjugate transposition. Then, the inverse discrete Fourier transform (IDFT) is given as

$$\begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{pmatrix} = \frac{1}{N} \mathbf{F}_N^* \begin{pmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{pmatrix} \quad (6)$$

Theorem 1. References [20,21] The matrix \mathbf{C} defined in (1) can be diagonalized by the DFT matrix \mathbf{F}_N , namely,

$$\mathbf{C} = \mathbf{F}_N^{-1} \text{diag}(\mathbf{F}_N \mathbf{C}) \mathbf{F}_N \quad (7)$$

where c is the first column of \mathbf{C} , i.e., $c = [c_0 \ c_1 \ \dots \ c_{N-1}]^T$.

Proof. It is easy to verify that the normalized eigenvectors of \mathbf{C} are given by

$$v_j = \frac{1}{\sqrt{N}} (1, \omega_j, \omega_j^2, \dots, \omega_j^{N-1})^T, j = 0, 1, \dots, N-1, \quad (8)$$

where $\omega_j = e^{\frac{2\pi i j}{N}}$ with $i = \sqrt{-1}$ is the N -th root of unity. The corresponding eigenvalues are then given by

$$\begin{aligned} \lambda_j &= c_0 + c_{N-1} \omega_j + c_{N-2} \omega_j^2 + \dots + c_1 \omega_j^{N-1} \\ &= c_0 \omega_j^{-0} + c_{N-1} \omega_j^N \omega_j^{-(N-1)} + \\ &\quad c_{N-2} \omega_j^N \omega_j^{-(N-2)} + \dots + c_1 \omega_j^N \omega_j^{-1} \\ &= \sum_{k=0}^{N-1} c_k \omega_j^{-k}, j = 0, 1, \dots, N-1. \end{aligned} \quad (9)$$

From (2), (3) and (9), we have

$$\begin{pmatrix} \lambda_0 \\ \vdots \\ \lambda_{N-1} \end{pmatrix} = \mathbf{F}_N \begin{pmatrix} c_0 \\ \vdots \\ c_{N-1} \end{pmatrix} = \mathbf{F}_N c \quad (10)$$

Using $\{v_j\}_{j=0}^{N-1}$ defined in Equation (8) as column vectors to form a $N \times N$ matrix \mathbf{U}_N

$$\mathbf{U}_N = [v_0, v_1, \dots, v_{N-1}], \quad (11)$$

then we have

$$\mathbf{C} = \mathbf{U}_N \begin{pmatrix} \lambda_0 & & \\ & \ddots & \\ & & \lambda_{N-1} \end{pmatrix} \mathbf{U}_N^*. \quad (12)$$

Furthermore, we have $\mathbf{U}_N^* = \frac{1}{\sqrt{N}} \mathbf{F}_N$, $\mathbf{U}_N = \sqrt{N} \mathbf{F}_N^{-1}$ from (4), (5) and (11). Then we have

$$\mathbf{C} = \mathbf{F}_N^{-1} \text{diag}(\mathbf{F}_N c) \mathbf{F}_N \quad (13)$$

□

Definition 3. Linear convolution [23]: A time-invariant linear operator L can be represented as a linear convolution. To be specific, we denote by $\delta[n]$ the discrete Dirac

$$\delta[n] = \begin{cases} 1 & \text{if } n = 0 \\ 0 & \text{if } n \neq 0 \end{cases}.$$

Any signal $f[n]$ can be decomposed as a sum of shifted Diracs:

$$f[n] = \sum_{p=-\infty}^{+\infty} f[p] \delta[n-p].$$

Let $L\delta[n] = h[n]$ be the discrete impulse response. Linearity and time invariance implies that

$$g[n] = Lf[n] = \sum_{p=-\infty}^{\infty} f[p] h[n-p] \stackrel{\text{def}}{=} f \star h[n], \quad (14)$$

where “ \star ” represents linear convolution.

Definition 4. Causality and Stability [23]: A discrete filter L is causal if $Lf[p]$ depends only on the values of $f[n]$ for $n \leq p$. The convolution formula (14) implies that $h[n] = 0$ if $n < 0$.

A discrete filter L is stable if any bounded input signal $f[n]$ produces a bounded output signal $Lf[n]$. One can verify that the filter is stable if and only if $h \in l^1(\mathbb{Z})$.

Proposition 1. Assume that the length of f and h are finite. To be specific, If $f = [f_0, f_1, \dots, f_{n-\beta-1}]^T$, and $h = [h_0, h_1, \dots, h_\beta]^T$, then the linear convolution of f and h defined in (14) can be written as

$$g = f \star h = \mathbf{H}f \quad (15)$$

where $\mathbf{H} \in R^{n \times (n-\beta)}$, and

$$\mathbf{H} = \begin{bmatrix} h_0 & & & & & \\ h_1 & h_0 & & & & \\ \vdots & & \ddots & & & \\ h_\beta & & & \ddots & & \\ & \ddots & & & h_0 & \\ & & \ddots & & h_1 & \\ & & & \ddots & \vdots & \\ & & & & h_\beta & \end{bmatrix}. \quad (16)$$

Proof. We refer the reader to [19] for details. \square

Suppose we wish to compute the polynomial product $c(x) = a(x) \cdot b(x)$, the ordinary product expression for the coefficients of $c(x)$ involves a linear convolution.

Definition 5. Circular convolution [19]: The circular convolution of a signal $\{x[k]\}_{k=0}^{N-1}$ with a signal $\{c[k]\}_{k=0}^{N-1}$ is defined as a matrix vector multiplication as follows.

$$y \stackrel{\text{def}}{=} x \otimes c = \mathbf{C} \cdot \begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-1] \end{bmatrix} \quad (17)$$

where \mathbf{C} is defined in equation (1), and “ \otimes ” represents circular convolution.

Proposition 2. The computation of circular convolution can be realized by using FFT-based fast method. In other word, Equation (17) can be written as

$$y = \mathbf{F}_N^{-1} \text{diag}(\mathbf{F}_N c) \mathbf{F}_N x. \quad (18)$$

Proof. The matrix involved in (17) is a circulant matrix, thus (18) is obtained by using Theorem 1. \square

The equivalent condition of circular convolution and linear convolution will be given in the following theorem.

Theorem 2. Assume $f = [f_0, f_1, \dots, f_{n-\beta-1}]^T$, $h = [h_0, h_1, \dots, h_\beta]^T$, $\tilde{f} = [f_0, f_1, \dots, f_{n-\beta-1}, \underbrace{0, \dots, 0}_\beta]^T$, $\tilde{h} = [h_0, h_1, \dots, h_\beta, \underbrace{0, \dots, 0}_{n-\beta-1}]^T$, then linear convolution $g = f \star h$ of f and h defined in (14) can be computed as circular convolution $\tilde{f} \otimes \tilde{h}$ of \tilde{f} and \tilde{h} , namely,

$$g = f \star h = \tilde{f} \otimes \tilde{h} = \mathbf{F}_N^{-1} (\mathbf{F}_N \tilde{h} \cdot \mathbf{F}_N \tilde{f}). \quad (19)$$

where \mathbf{F}_N , \mathbf{F}_N^{-1} are defined by (4), (5) respectively; “ \cdot ” means componentwise product of two vectors.

Proof. From Proposition (1), we get $g = \mathbf{H}f$, where $\mathbf{H} \in \mathbb{R}^{n \times (n-\beta)}$ is defined in (16). Now extend \mathbf{H} to a circulant matrix [21]

$$\mathbf{C} = [\mathbf{H}, \hat{\mathbf{H}}] = \begin{bmatrix} h_0 & & & h_\beta & & h_1 \\ h_1 & h_0 & & & \ddots & \vdots \\ \vdots & & \ddots & & & h_\beta \\ h_\beta & & & \ddots & & \\ & \ddots & & h_0 & & \\ & & \ddots & h_1 & h_0 & \\ & & & \ddots & \vdots & \ddots \\ & & & h_\beta & h_{\beta-1} & \cdots & h_0 \end{bmatrix}, \quad (20)$$

thus the first column of \mathbf{C} is \tilde{h} . Therefore,

$$g = \mathbf{H}f = [\mathbf{H}, \hat{\mathbf{H}}]\tilde{f} = \mathbf{C}\tilde{f} = \tilde{f} \otimes \tilde{h} = \mathbf{F}_N^{-1}(\mathbf{F}_N \tilde{h} * \mathbf{F}_N \tilde{f})$$

by using Definition 5 and Proposition 2. \square

From Theorem 2, the algorithm of linear convolution of $g = f \star h$ is given as follows (See Algorithm 1).

Algorithm 1: FFT-based fast method for linear convolution.

Input:

the signal, $f[k], k = 0, 1, \dots, n - \beta - 1$

the signal, $h[k], k = 0, 1, \dots, \beta$

Output:

the linear convolution of f and h , $g[k], k = 0, 1, \dots, n - 1$

1. Set $n = \text{length}(f) + \text{length}(h) - 1$;
 2. Appending an array of zeros to the end of h to obtain \tilde{h} so that the length of \tilde{h} is n .
 3. Appending an array of zeros to the end of f to obtain \tilde{f} so that the length of \tilde{f} is n
 4. $g = \text{ifft}(\text{fft}(\tilde{h}) * \text{fft}(\tilde{f}))$
-

Additionally, the asymptotic time complexity is $\mathcal{O}(n \log_2(n))$ for Algorithm 1. Note the condition $n = \text{length}(f) + \text{length}(h) - 1$ plays a pivotal role in the equivalence of linear convolution and circular convolution. This condition will also be used in the following part of this paper.

Definition 6. Equivalent Condition of Linear Convolution and Circular Convolution: Let $f, h, \tilde{f}, \tilde{h}$ be defined in Theorem 2. The condition $\text{length}(\tilde{h}) = \text{length}(\tilde{f}) = \text{length}(f) + \text{length}(h) - 1$ is called Equivalent Condition of Linear Convolution and Circular Convolution.

3. A Class of Algorithms for Continuous Wavelet Transform

3.1. Time-Domain Algorithm for CWT

Definition 7. Continuous wavelet transform: The continuous wavelet transform of a function $f(t)$ at a scale $a \in \mathbb{R}^+$ and translational value $b \in \mathbb{R}$ is expressed by the following integral [17]

$$W_f(a, b) \stackrel{\text{def}}{=} \int_{-\infty}^{+\infty} f(t) \frac{1}{a} \bar{\psi}\left(\frac{t-b}{a}\right) dt \quad (21)$$

where $\psi(t)$ is a continuous function called the mother wavelet and the overline represents the operation of complex conjugate.

For real-life applications, the length of $f(t)$ is finite and the support of $\psi(t)$ is compact. Without loss of generality, assume the sampling rates of the signal and the wavelet are equal to 1, then Equation (21) can be written as

$$W_f^a[m] = \sum_{k=-\infty}^{+\infty} f(k) \frac{1}{a} \bar{\psi}\left(\frac{k-m}{a}\right), \quad \text{where } m \in \mathbb{Z}. \quad (22)$$

Assume the sampled signal is $\{f[i]\}_{i=0}^{N-1}$ and the support of the mother wavelet $\psi(t)$ is $[-T, T]$. If define

$$\tilde{\psi}_a(t) \stackrel{\text{def}}{=} \frac{1}{a} \bar{\psi}\left(\frac{aT-t}{a}\right) \quad (23)$$

then the support of $\tilde{\psi}_a(t)$ is $[0, 2aT]$. Now, we can get the following result.

Theorem 3. Assume the length of $f(t)$ is finite and the support of $\psi(t)$ is compact, and the sampling rates of the signal and the wavelet are equal to 1, then CWT of $\{f[i]\}_{i=0}^{N-1}$ can be written as

$$W_f^a[m] = \sum_{k=0}^{N-1} f(k) \tilde{\psi}_a(m + aT - k) = f \star \tilde{\psi}_a[m + aT], \text{ where } m = 0, 1, \dots, N-1, \quad (24)$$

where we assume aT is an integer for the convenience of analysis. Furthermore, the corresponding filter is causal and stable.

Proof. By using Proposition 1 and Equations (23) and (24) is deduced from (22). Furthermore, define

$$h_a[i] = \tilde{\psi}_a(i), i \in \mathbb{Z} \quad (25)$$

then the discrete filter $\{h_a[i]\}_{i \in \mathbb{Z}}$ is causal and stable since $h_a[i] = \tilde{\psi}_a(i) = 0$ if $i < 0$ and $h_a \in l^2(\mathbb{Z})$. Furthermore, Equation (24) can be written as

$$W_f^a[m] = f \star h_a[m + aT], \quad \text{where } m = 0, 1, \dots, N-1. \quad (26)$$

□

In brief, $\{W_f^a[i]\}_{i=0}^{N-1}$ are just the $\{f \star h_a[i]\}_{i=aT}^{aT+N-1}$ while $\{f \star h_a[i]\}_{i=0}^{2aT+N-1}$ can be implemented with linear convolution [17]. From Algorithm 1, we can get the algorithm of CWT (see Algorithm 2).

Algorithm 2: Time-domain fast algorithm for CWT.

Input:

the signal, $f[n], n = 0, 1, \dots, N-1$
 mother wavelet, $\psi(t)$
 scale, a
 T , where $[-T, T]$ is the support of $\psi(t)$.

Output:

wavelet coefficients, $W_f^a[i], i = 0, 1, \dots, N-1$

1. Let $\tilde{f} = [f_0, f_1, \dots, f_{N-1}, \underbrace{0, \dots, 0}_{2aT}]^T$;
 2. Let $\tilde{\psi}_a(t) = \frac{1}{a} \bar{\psi}\left(\frac{aT-t}{a}\right)$, where the overline represents the operation of complex conjugate;
 3. Let $h_a[n] = \tilde{\psi}_a(n), n = 0, 1, \dots, 2aT$;
 4. Let $\tilde{h}_a = [h_a[0], h_a[1], \dots, h_a[2aT], \underbrace{0, \dots, 0}_{N-1}]^T$;
 5. $tem_a[i] = \text{ifft}(\text{fft}(\tilde{f}) \cdot \text{fft}(\tilde{h}_a)), i = 0, 1, \dots, 2aT + N - 1$;
 6. $W_f^a[i] = tem_a[aT + i], i = 0, 1, \dots, N - 1$.
-

3.2. Frequency-Domain Algorithm for CWT

The algorithm can be further optimized by taking advantage of the analytic expression of $\hat{\psi}(\omega)$, the Fourier transform of the mother wavelet $\psi(t)$. In fact, the wavelet transform defined in (21) can also be written as a frequency integration by applying the Fourier Parseval formula [23].

$$W_f(a, b) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \hat{f}(\omega) \bar{\hat{\psi}}(a\omega) e^{ib\omega} d\omega \quad (27)$$

Assume the sampled signal is $\{f[n]\}_{n=0}^{N-1}$. Discretizing (27) and considering the periodic property of discrete Fourier transform yields

$$\begin{aligned} W_f^a[n] &= \frac{1}{M} \sum_{k=-\frac{M}{2}+1}^{\frac{M}{2}} \hat{f}\left(\frac{2\pi}{M}k\right) \bar{\hat{\psi}}\left(\frac{2\pi}{M}ka\right) e^{i\frac{2\pi}{M}kn} \\ &= \frac{1}{M} \sum_{k=0}^{M-1} \hat{f}\left(\frac{2\pi}{M}k\right) \bar{\hat{\psi}}\left(\frac{2\pi}{M}ka\right) e^{i\frac{2\pi}{M}kn}, \\ \text{where } n &= 0, 1, \dots, M-1. \end{aligned} \quad (28)$$

It is seen by comparing (28) and (18) that (28) represents a circular convolution. By considering Equivalent Condition of Linear Convolution and Circular Convolution, It is reasonable to let

$$M = 2aT + N. \quad (29)$$

In this case, the values of $\{\hat{f}(\frac{2\pi}{M}k)\}_{k=0}^{M-1}$ can be computed as the discrete Fourier transform of $\tilde{f} = [f_0, f_1, \dots, f_{N-1}, \underbrace{0, \dots, 0}_{M-N}]^T$. As for the computation of $\{\bar{\hat{\psi}}(\frac{2\pi}{M}ka)\}_{k=0}^{M-1}$, we can make use of the analytic expression of $\hat{\psi}(\omega)$. For example, the Morlet wavelet is defined as [24]

$$\psi(t) \stackrel{\text{def}}{=} \sqrt{\frac{2}{\pi\sigma^2}} e^{-\frac{t^2}{2\sigma^2}} e^{i\eta t}, \quad (30)$$

then

$$\bar{\hat{\psi}}(\omega) = \hat{\psi}(\omega) = 2e^{-\frac{\sigma^2(\omega-\eta)^2}{2}}, \quad (31)$$

where σ^2 is shape parameter, and η is center frequency. Furthermore, the Morlet wavelet is approximately analytic and therefore $\bar{\hat{\psi}}(\frac{2\pi}{M}ka) \approx 0$ for $k < 0$. So

$$\bar{\hat{\psi}}\left(\frac{2\pi}{M}ka\right) \approx 0, \quad \text{for } k > \frac{M}{2}, \quad (32)$$

by the periodic property of discrete Fourier transform. Equation (32) will be used to design algorithm of CWT (see the 4th–6th steps of Algorithms 3).

Definition 8. Equivalent Condition of Time-domain and Frequency-domain Algorithms of CWT: $M = 2aT + N$ in Equation (29) is called Equivalent Condition of Time-domain and Frequency-domain Algorithms of CWT.

Remark 1. In freeware such as Wavelab, the parameter M in Equation (29) takes value N , where N is the length of signal. In this case, $\{\hat{f}(\frac{2\pi}{M}k)\}_{k=0}^{M-1}$ can be obtained as the discrete Fourier transform of the data. Furthermore, the length of the result of **ifft** is just N . However, the method will produce artificial periodicity, which is called edge effect by Torrence [18], in the wavelet coefficients if signal is not periodic. In order to limit

the edge effect, in [18], the data is padded with sufficient zeros before doing the CWT, then the first N coefficients of the corresponding convolution are just the coefficients of the CWT. Nevertheless, the reason for this is not answered in previous papers in the author's knowledge. Put another way. Why the last step of Algorithm 3 is dissimilar from the last step of Algorithm 2? The answer will be given in the following Theorem 4.

Lemma 1. The filter used in Equation (27) is

$$\tilde{\psi}_a(t) = \frac{1}{a} \tilde{\psi}\left(\frac{-t}{a}\right). \quad (33)$$

Proof. We refer the reader to [23] for details. \square

Theorem 4. The wavelet coefficients of the CWT of $\{f[n]\}_{n=0}^{N-1}$ is the first N coefficients of (28). While the length of the total coefficients of (28) is M .

Proof. The filter used in Theorem 3 is $\tilde{\psi}_a$ defined in (23); therefore, $\tilde{\psi}_a(t - aT) = \tilde{\psi}_a(t)$. Define $L\tilde{\psi}_a = f \star \tilde{\psi}_a$, then $L\tilde{\psi}_a = f \star \tilde{\psi}_a$. Denote $\tilde{\tilde{\psi}} = \tilde{\psi}_a$, $\tilde{\tilde{\psi}} = \tilde{\psi}_a$, $\tilde{\tilde{\psi}}_{aT} = \tilde{\tilde{\psi}}(t - aT) = \tilde{\tilde{\psi}}$ for the simplicity of notation for this moment. By the property of time-invariant of L ,

$$L\tilde{\tilde{\psi}}[n] = L\tilde{\tilde{\psi}}_{aT}[n] = L\tilde{\tilde{\psi}}[n - aT];$$

therefore

$$L\tilde{\tilde{\psi}}[n + aT] = L\tilde{\tilde{\psi}}[n].$$

That is to say

$$\{L\tilde{\tilde{\psi}}_a[n + aT]\}_{n=0}^{N-1} = \{L\tilde{\tilde{\psi}}_a[n]\}_{n=0}^{N-1}.$$

From Theorem 3, $\{L\tilde{\tilde{\psi}}_a[n + aT]\}_{n=0}^{N-1}$ is the wanted wavelet coefficients; therefore $\{L\tilde{\tilde{\psi}}_a[n]\}_{n=0}^{N-1} = \{f \star \tilde{\tilde{\psi}}_a[n]\}_{n=0}^{N-1}$ is the wanted wavelet coefficients. \square

Theorem 5. Assume that (Without loss of generality, aT is assumed to be an integer.)

$$\begin{aligned} L_m[n] &= \frac{1}{M} \sum_{k=0}^{M-1} \hat{f}\left(\frac{2\pi}{M}k\right) \tilde{\tilde{\psi}}\left(\frac{2\pi}{M}ka\right) e^{-i\frac{2\pi}{M}kmaT} e^{i\frac{2\pi}{M}kn}, \\ n &= 0, 1, \dots, M-1; m = 0, 1, 2, \end{aligned} \quad (34)$$

then

$$\{W_f^a[n]\}_{n=0}^{N-1} = \{L_0[n]\}_{n=0}^{N-1} = \{L_1[n]\}_{n=aT}^{aT+N-1} = \{L_2[n]\}_{n=M-N}^{M-1}.$$

Proof. If $m = 0$, Equation (34) is simply Equation (28), the conclusion is deduced from Theorem 4.

Since

$$\begin{aligned} F[\tilde{\tilde{\psi}}_a](\omega) &= \int_{-\infty}^{+\infty} \frac{1}{a} \tilde{\tilde{\psi}}\left(\frac{aT-t}{a}\right) e^{-it\omega} dt \\ &= e^{-iaT\omega} \tilde{\tilde{\psi}}(a\omega), \end{aligned} \quad (35)$$

thus $L_1[n] = f \star \tilde{\tilde{\psi}}_a$, the conclusion is deduced from Theorem 3. If define $\psi_a^{(3)} = \frac{1}{a} \tilde{\tilde{\psi}}\left(\frac{2aT-t}{a}\right)$, the conclusion for the $m = 2$ case can also be proved in the same way. \square

Now, Frequency-domain algorithm of CWT with Morlet wavelet as the mother wavelet is given as follows.

Algorithm 3: Frequency-domain fast algorithm for CWT (In real application, the case $m = 0$ is enough for the calculation of CWT).

Input:

the signal, $f[n], n = 0, 1, \dots, N - 1$

scale, a

$\hat{\psi}(\omega)$, the Fourier transform of the Morlet wavelet $\psi(t)$

T , where $[-T, T]$ is the support of $\psi(t)$.

Output:

wavelet coefficients, $W_f^a[i], i = 0, 1, \dots, N - 1$

1. Let $\tilde{f} = [f_0, f_1, \dots, f_{N-1}, \underbrace{0, \dots, 0}_{2aT}]^T$;
 2. Define $\varphi_m(\omega) = e^{-imaT\omega} \overline{\hat{\psi}(a\omega)}$, $m = 0, 1, 2$, where the overline represents the operation of complex conjugate;
 3. Let $M = 2aT + N$;
 4. If M is even, Let $\hat{h}_a = [\varphi_m(\frac{2\pi}{M} \cdot 0), \varphi_m(\frac{2\pi}{M} \cdot 1), \dots, \varphi_m(\frac{2\pi}{M} \cdot \frac{M}{2}), \underbrace{0, \dots, 0}_{\frac{M}{2}-1}]^T$;
 5. else $\hat{h}_a = [\varphi_m(\frac{2\pi}{M} \cdot 0), \varphi_m(\frac{2\pi}{M} \cdot 1), \dots, \varphi_m(\frac{2\pi}{M} \cdot \frac{M-1}{2}), \underbrace{0, \dots, 0}_{\frac{M-1}{2}}]^T$;
 6. end if
 7. $tem_a[i] = ifft(fft(\tilde{f}) \cdot (\hat{h}_a)), i = 0, 1, \dots, 2aT + N - 1$;
 8. switch(m)

case 0	$W_f^a[i] = tem_a[i], \quad i = 0, 1, \dots, N - 1$;
case 1	$W_f^a[i] = tem_a[i + aT], \quad i = 0, 1, \dots, N - 1$;
case 2	$W_f^a[i] = tem_a[i + 2aT], \quad i = 0, 1, \dots, N - 1$.
-

Remark 2. Note that the previous data preparation method takes $M = 2^q$, where $q = \min_{n \in \mathbb{Z}} \{2^n > N\}$ [17,18]. However, this method may fail for some real life data. We propose $M = N + 2aT$, Equivalent Condition of Time-domain and Frequency-domain Algorithms of CWT, then the edge effect, defined in [18], that may occur in JLAB, Wavelab and [25] can be avoided. See Figures 6 and 7 for details.

4. Numerical Experiments

The experiments are conducted on two types of data, one for synthetic data, another for real-life data. Entropy can be used to measure the sparsity of wavelet coefficients [17]. In order to define entropy, $\{|W_f(a_i, b_j)|^2\}_{i,j}$ are rearranged as $\{c_k\}_{k=1, \dots, M}$, then are normalized to obtain:

$$d_k = \frac{c_k}{\sum_{k=1}^M c_k}. \quad (36)$$

The wavelet entropy is calculated as

$$Entropy = - \sum_{k=1}^M d_k \log d_k, \quad (37)$$

with the convention $0 \log 0 = 0$ by definition.

Experiments for Data 1: Data 1 is a synthetic signal with length N . The first half contains sinusoidal signal superimposition with three different frequencies, namely $a_4 \sin(30\pi t) + a_3 \sin(60\pi t) + a_1 \sin(120\pi t)$; The latter half contains the 60 Hz sinusoidal signal with amplitude a_2 , namely $a_2 \sin(120\pi t)$, where $a_4 = 1, a_3 = 1.2, a_1 = 1.4, a_2 = 0.6$, and the sampling frequency is 400 Hz. Data 1 with length $N = 1024$ is presented in Figure 1a. Figure 1b, the absolute values of CWT coefficients of Figure 1a, computed with the *awt* function in Wavelab, where the shape parameter

$\sigma^2 = 1$ and the center frequency $\eta = 8$, manifests edge effects. Figure 1c without edge effects is computed by using the case 0 of Algorithm 3.

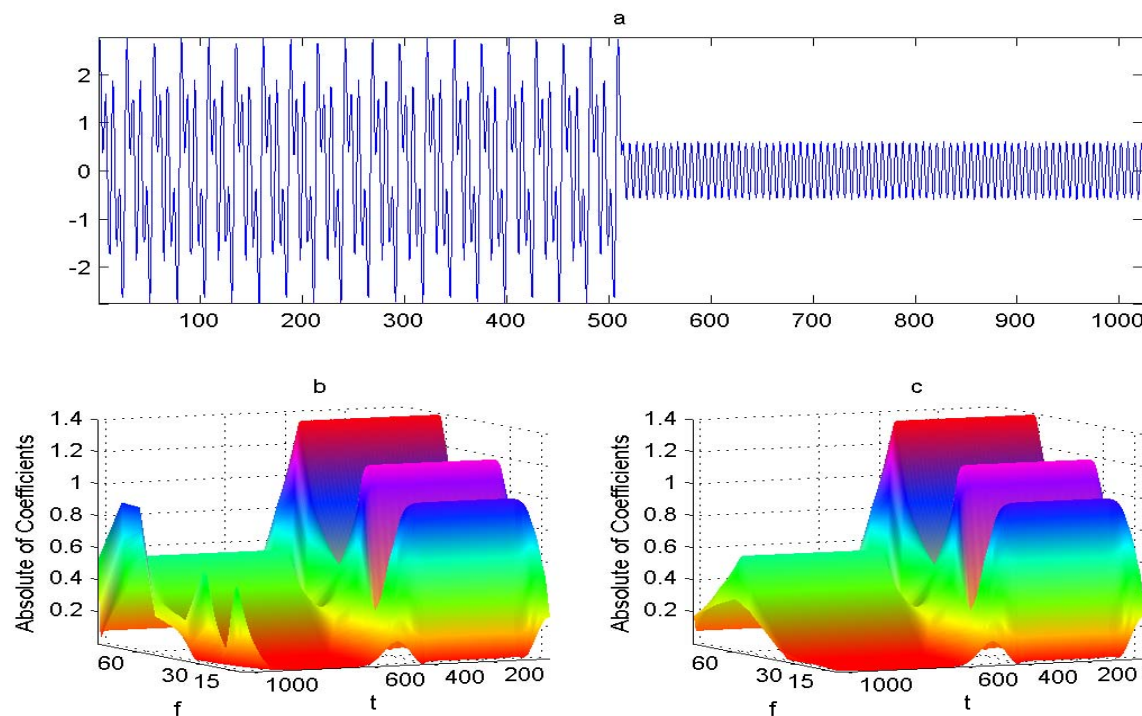


Figure 1. (a) is Data 1 with $N = 1024$. (b) the absolute values of CWT coefficients of (a), computed with the *awt* function in Wavelab, where the shape parameter $\sigma^2 = 1$ and the center frequency $\eta = 8$, manifests edge effects. (c) without edge effects is computed with Algorithm 3.

Table 1 gives the computational results of Data 1 with wavelet parameter $\sigma^2 = 1, \eta = 8$ by using different methods. “*cwt*” means the *cwt* function in wavelet toolbox of Matlab. “Zhao’s method” is an improved version of *cwt* [26]. “Direct” means the computation method of linear convolution by using Equation (15). If the length of signal is far larger than the length of the filter, an “Overlap-add” procedure for the calculation of linear convolution is faster than Direct method and Algorithm 1 [23]. The “wavelet entropy” shown in Table 1 can measure the sparsity of wavelet coefficients [17]. “Err_i” means the relative error of a_i and the corresponding maximum amplitude of CWT coefficients.

From Table 1, we know that the precision, wavelet entropy of “Direct”, Algorithms 2 and 3, “Overlap-add” are almost the same and are more optimal to *cwt* function in toolbox of Matlab. In fact, the precision of two methods in [17], “FFT based method” and the proposed method is almost the same. This phenomenon indicates some equivalence of these two methods. As a matter of fact, these two methods can be categorized as Algorithms 2 and 3 respectively.

Figure 2 is computed by using the case 1 of Algorithm 3. The wavelet coefficients of Figure 2b can exactly characterize the time-frequency local properties of data of Figure 1a. At the same time, Figure 2a,c fail to do so. In fact, it is known from the case 1 of Algorithm 3, the wavelet coefficients should be chosen as the middle part of the coefficients of the convolution. However, the wavelet coefficients of Figure 2a,c are respectively chosen as the first N and the last N part of the coefficients of the convolution.

Table 1. Comparisons between the different CWT methods for Data 1.

Algorithm	N	CPU	Entropy	Err_1	Err_2	Err_3	Err_4
cwt	1024	0.2468	8.7125	3.3605×10^{-2}	3.5635×10^{-2}	-1.4622×10^{-2}	-4.0837×10^{-2}
	2048	0.2560	9.3815	3.3605×10^{-2}	3.5635×10^{-2}	-1.4622×10^{-2}	-4.0837×10^{-2}
Zhao's	1024	0.0495	8.7068	-1.0144×10^{-2}	3.6888×10^{-3}	-1.6761×10^{-2}	-2.8041×10^{-2}
	2048	0.0562	9.3762	-1.0144×10^{-2}	3.6888×10^{-3}	-1.6755×10^{-2}	-2.8041×10^{-2}
Direct	1024	0.4724	8.7097	-2.0333×10^{-4}	-1.8504×10^{-15}	-1.9882×10^{-4}	4.9072×10^{-14}
	2048	0.8697	9.3784	-2.0333×10^{-4}	-7.0314×10^{-15}	-1.9767×10^{-4}	-9.6064×10^{-11}
Overlap-Add	1024	0.0291	8.7083	-2.0333×10^{-4}	-1.8504×10^{-15}	-1.9882×10^{-4}	4.8517×10^{-14}
	2048	0.0359	9.3777	-2.0333×10^{-4}	-6.4763×10^{-15}	-1.9767×10^{-4}	-9.6065×10^{-11}
Algorithm 2	1024	0.0204	8.7083	-2.0333×10^{-4}	-1.8504×10^{-15}	-1.9882×10^{-4}	4.8517×10^{-14}
	2048	0.0291	9.3777	-2.0333×10^{-4}	-6.4763×10^{-15}	-1.9767×10^{-4}	-9.6065×10^{-11}
Algorithm 3	1024	0.0155	8.7083	-2.0333×10^{-4}	-2.4055×10^{-15}	-1.9882×10^{-4}	4.7851×10^{-14}
	2048	0.0223	9.3777	-2.0333×10^{-4}	-7.0314×10^{-15}	-1.9767×10^{-4}	-9.6065×10^{-11}

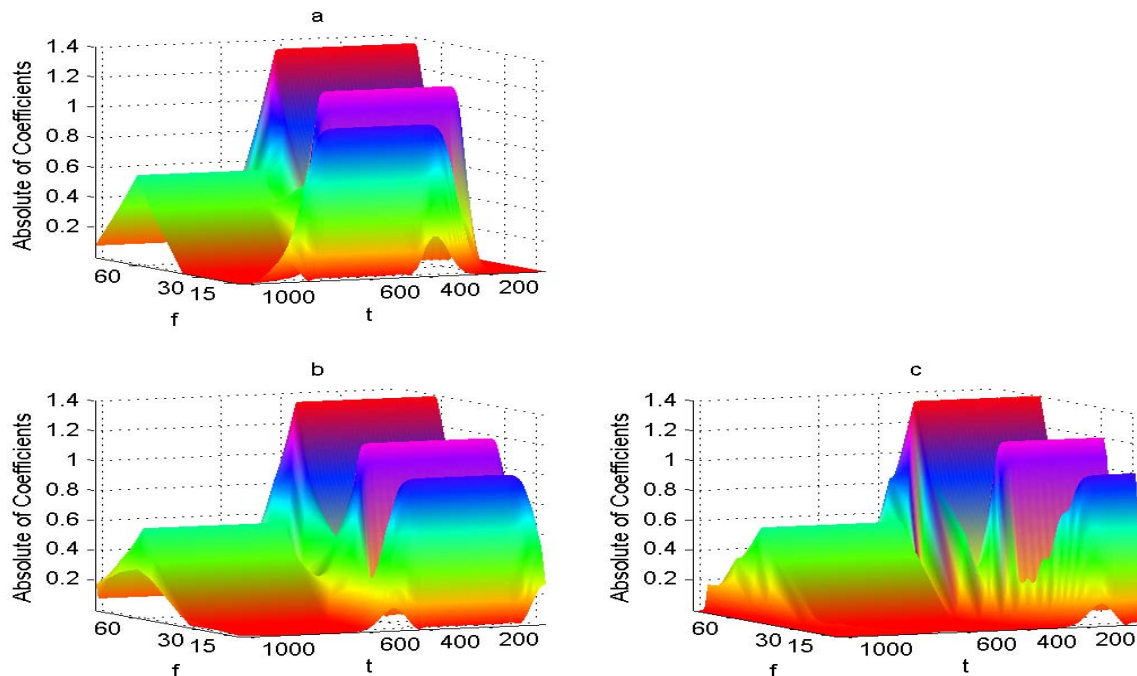


Figure 2. Figure 2 is computed by using the case 1 of Algorithm 3. The wavelet coefficients of (b) can exactly characterize the time-frequency local properties of data of (a). At the same time, (a,c) fail to do so.

Experiments for Data 2: Data 2 is the annual average temperature from 1 January 1951 to 31 December 2010 for No. 50978 weather station, Heilongjiang, China. Figures 3–5, computed with different methods of CWT, namely, Algorithm 2, Zhao's method, *cwt* function in wavelet toolbox of Matlab, are the contours of the real part of coefficients of CWT for Data 2. The Morlet wavelet parameters are $\sigma^2 = 1, \eta = 5$. The coefficients with negative values are plotted with a dotted, blue line, while the coefficients with positive values are plotted with a dashed, red line. It is obvious from these figures that the precision of Algorithm 2 is superior to that of Zhao's method and *cwt* function of Matlab.

It is seen from Table 2 that the wavelet entropy for Algorithm 2 is the smallest of all the methods. Therefore, Algorithm 2 is the most suitable method for this temperature data.

Figures 6 and 7 are calculated with different data preparation methods. To be specific, Figure 6 is computed with the frequency-domain method of CWT with data preparation method given in

previously published papers, namely, $M = 2^q, q = \min_{n \in \mathbb{Z}} \{2^n > N\}$ while Figure 7 is computed with the case 0 of Algorithm 3 with $M = N + 2aT$. Note the different structures of contours in the upper edge of these two figures. The contour is not closed in the upper edge of Figure 6, which means that there will be a valley and a peak in the left and right part of this edge. So the edge effect is obvious in Figure 6.

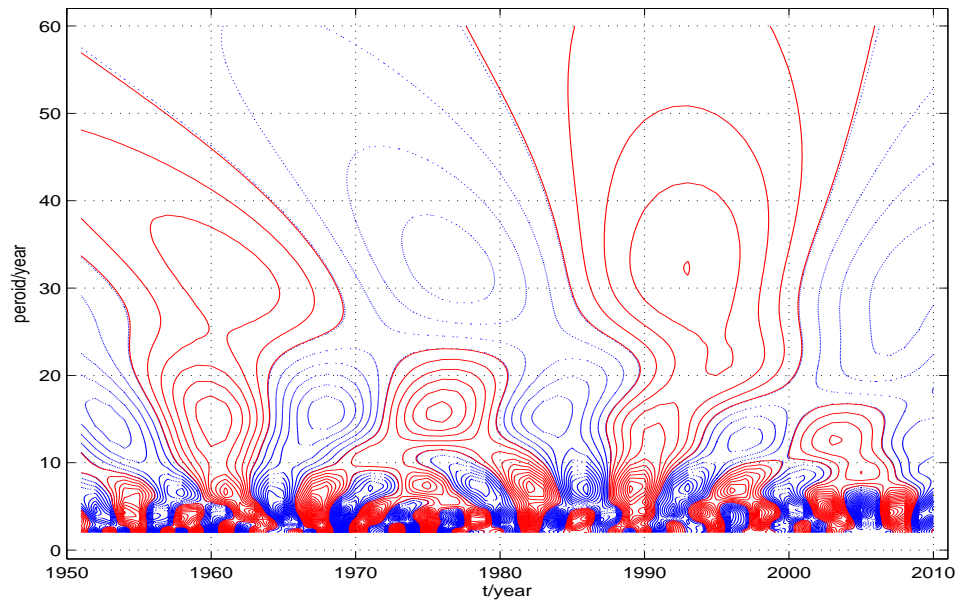


Figure 3. This contour is computed by using Algorithm 2 with the Morlet wavelet parameter $\sigma^2 = 1, \eta = 5$ for Data 2.

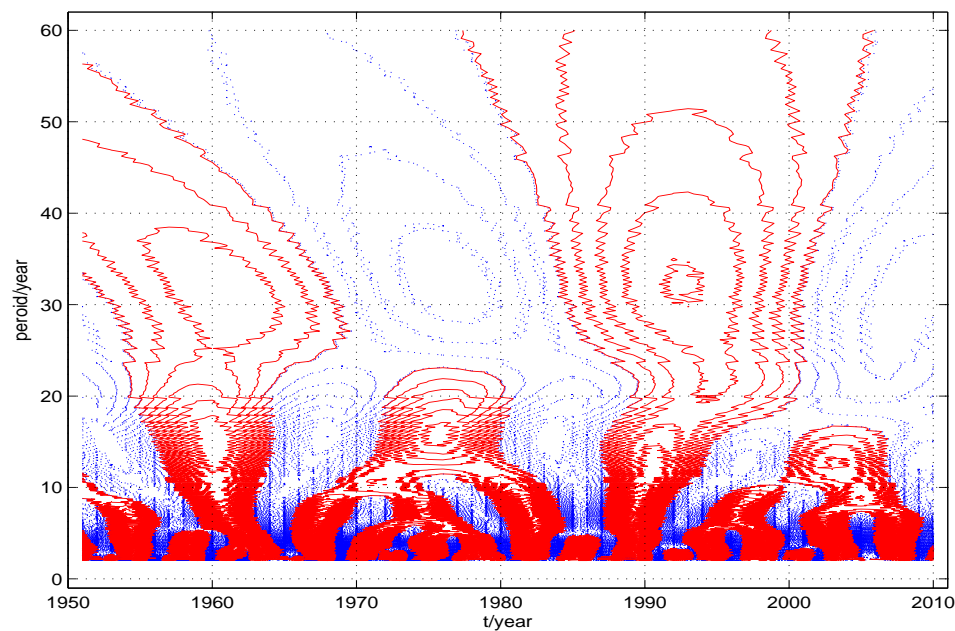


Figure 4. This contour is computed by using Zhao's method with the Morlet wavelet parameter $\sigma^2 = 1, \eta = 5$ for Data 2.

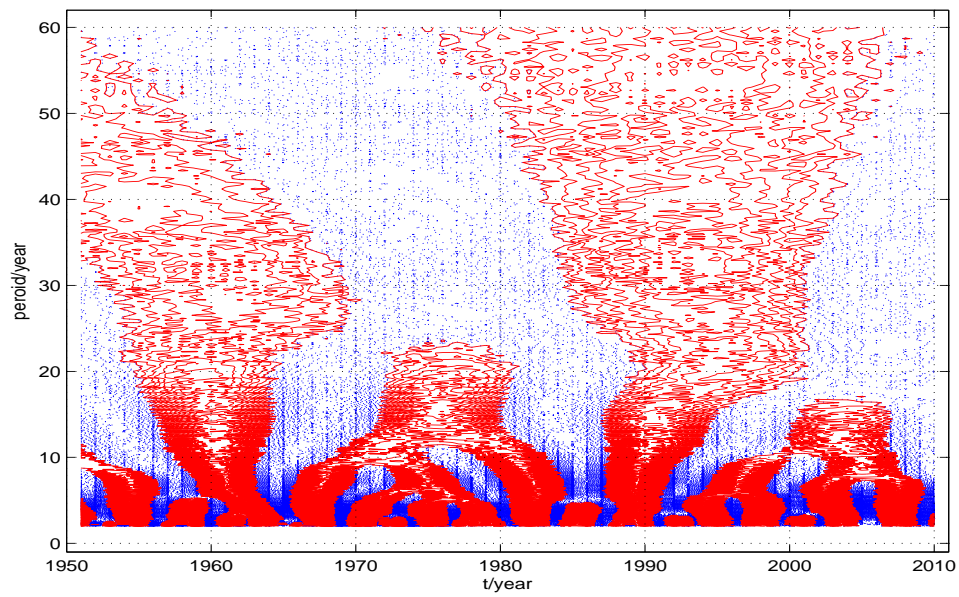


Figure 5. This contour is computed by using CWT function in wavelet toolbox of Matlab with the Morlet wavelet parameter $\sigma^2 = 1, \eta = 5$ for Data 2.

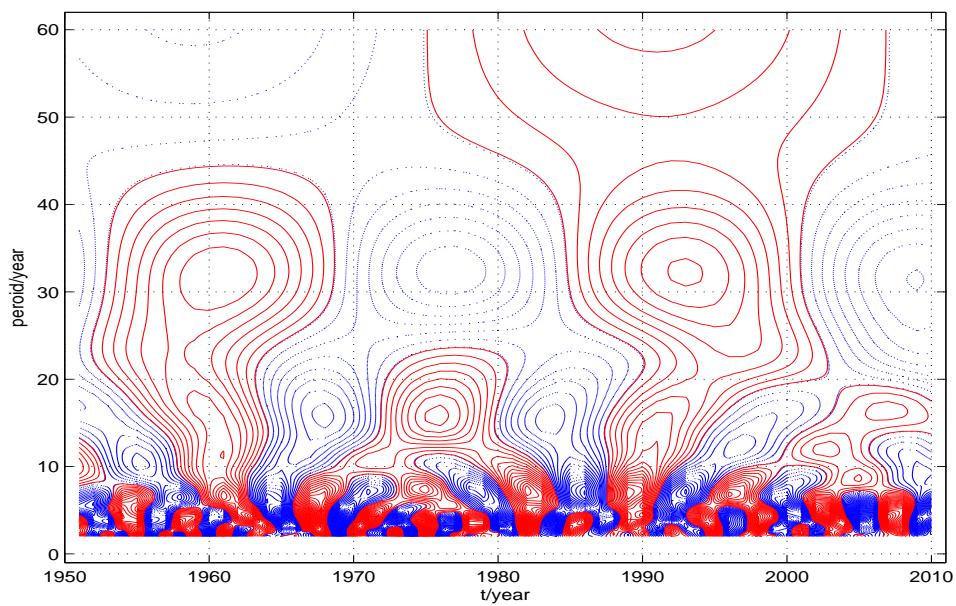


Figure 6. This contour, computed with frequency-domain method of CWT with data preparation method given in previously published papers, namely, $M = 2^q, q = \min_{n \in \mathbb{Z}} \{2^n > N\}$, manifests edge effect.

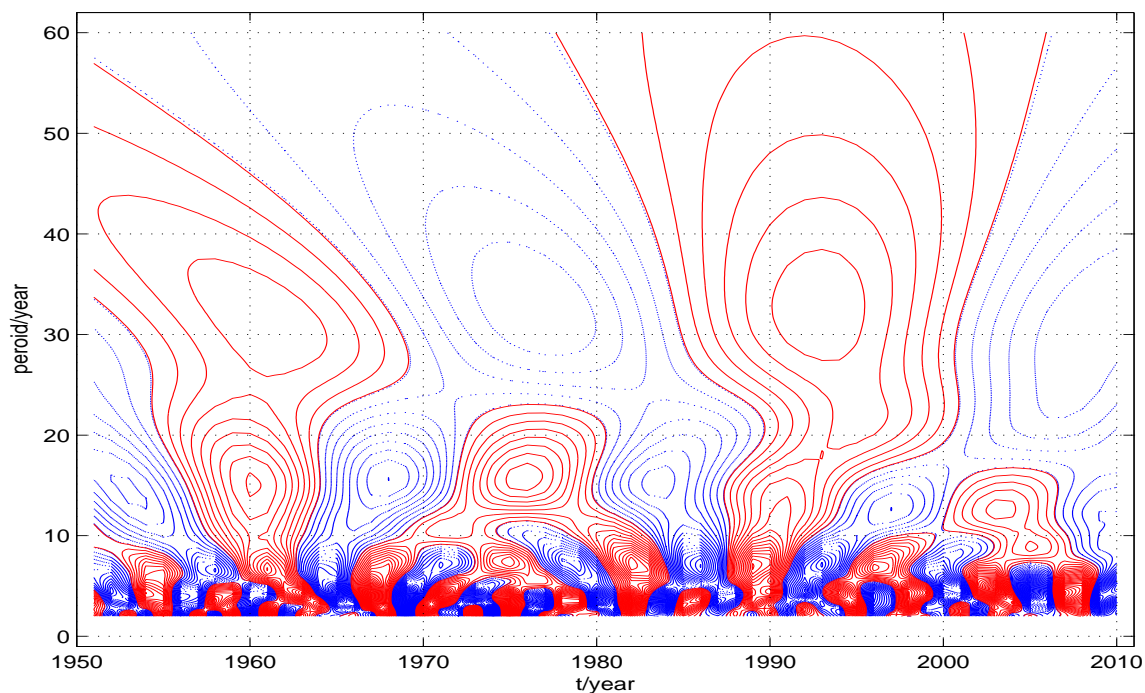


Figure 7. This contour, computed with the case 0 of Algorithm 3 with $M = N + 2aT$, manifests no edge effect.

Table 2. The wavelet entropy corresponding to different wavelet transform methods.

σ_2	η	CWT	Zhao's Method	Algorithm 2	Algorithm 3
1	5	9.5190	9.4150	9.4144	9.4534
1	6	9.5051	9.4044	9.4042	9.4332
1	7	9.4869	9.3897	9.3893	9.4117
1	8	9.4677	9.3713	9.3709	9.3892
1.5	5	9.5012	9.4030	9.4026	9.4304
1.5	6	9.4797	9.3832	9.3831	9.4037
1.5	7	9.4545	9.3601	9.3598	9.3762
1.5	8	9.4323	9.3377	9.3374	9.3513

5. Conclusions

The continuous wavelet transform of a signal with finite length for a fixed scale is considered as a linear time-invariant operator. Furthermore, the filter with causal and stability is constructed. The algorithm of linear convolution constitutes a unifying framework to the continuous wavelet transform methods previously published in [17]. The precision of these methods based on this framework is almost the same, no matter what method is used, and higher than other methods that use the approximation of wavelet, for example, the *cwt* function in wavelet toolbox of Matlab. The edge effect is also easily handled by using Equivalent Condition of Time-domain and Frequency-domain Algorithms of CWT.

The algorithms of CWT consist of two methods, time-domain method and frequency-domain method. For time-domain method, by constructing the causal filter $\tilde{\psi}_a(t)$, we know the wavelet coefficients are just the middle part of the corresponding linear convolution. For frequency-domain method, by exploring the relationship of $\tilde{\psi}_a(t)$ and $\tilde{\tilde{\psi}}_a(t)$, we know the wavelet coefficients are just the first N coefficients of the corresponding convolution. Furthermore, by constructing the different filters, the wavelet coefficients can be the first N coefficients, or the middle N coefficients, or the last N coefficients of the corresponding convolution for frequency-domain method.

There are three methods for the calculation of linear convolution. The first one is to directly implement the definition of linear convolution. The second one is known as the FFT-based method, for example, Algorithm 1. Lastly, if the length of the signal is far larger than the length of the filter, an overlap-add procedure for the calculation of linear convolution is faster than the previous two methods [23]. How to combine the Frequency-domain method and an overlap-add procedure for the computation of CWT is a question for future research.

Acknowledgments: The work is supported by the Natural Science Foundation of Jiangxi Province, China (Grant No. 20161BAB201017), the Technology Plan Project of Jiangxi Provincial Education Department (No. GJJ160758), and the Doctoral research startup project of Jinggangshan University (No. JZB11002).

Author Contributions: Hua Yi, Shi-You Xin and Jun-Feng Yin conceived and designed the experiments; Jun-Feng Yin wrote the introduction part of this paper; Shi-You Xin wrote the experimental part; Hua Yi wrote the rest parts of this paper.

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

References

- Huang, N.E.; Shen, Z.; Long, S.R.; Wu, M.C.; Shih, H.H.; Zheng, Q.; Yen, N.C.; Chi, C.T.; Liu, H.H. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proc. Math. Phys. Eng. Sci.* **1998**, *454*, 903–995.
- Daubechies, I.; Heil, C. Ten Lectures on Wavelets. *Comput. Phys.* **1992**, *6*, 1671, doi:10.1063/1.4823127.
- Gilles, J. Empirical Wavelet Transform. *IEEE Trans. Signal Process.* **2013**, *61*, 3999–4010.
- Amezquita-Sanchez, J.P.; Adeli, H. A new music-empirical wavelet transform methodology for time frequency analysis of noisy nonlinear and non-stationary signals. *Digit. Signal Process.* **2015**, *45*, 55–68.
- Daubechies, I.; Lu, J.; Wu, H.T. Synchrosqueezed wavelet transforms: An empirical mode decomposition-like tool. *Appl. Comput. Harmon. Anal.* **2011**, *30*, 243–261.
- Garcia-Perez, A.; Amezquita-Sanche, J.P.; Dominguez-Gonzalez, A.; Sedaghat, R.; Osornio-Rio, R.; Romero-Troncos, R.J. Fused empirical mode decomposition and wavelets for locating combined damage in a truss-type structure through vibration analysis. *J. Zhejiang Univ. Sci. A* **2013**, *14*, 615–630.
- Boashash, B.; Khan, N.A.; Ben-Jabeur, T. Time frequency features for pattern recognition using high-resolution TFDs: A tutorial review. *Digit. Signal Process.* **2015**, *40*, 1–30.
- Glowacz, A. DC Motor Fault Analysis with the Use of Acoustic Signals, Coiflet Wavelet Transform, and K-Nearest Neighbor Classifier. *Arch. Acoust.* **2015**, *40*, 321–327.
- Chen, J.; Rostami, J.; Tse, P.W.; Wan, X. The Design of a Novel Mother Wavelet that is Tailor-made for Continuous Wavelet Transform in Extracting Defect-Related Features from Reflected Guided Wave Signals. *Measurement* **2017**, *110*, 176–191.
- Chamanzar, A.; Shabany, M.; Malekmohammadi, A.; Mohammadinejad, S. Efficient Hardware Implementation of Real-Time Low-Power Movement Intention Detector System Using FFT and Adaptive Wavelet Transform. *IEEE Trans. Biomed. Circuits Syst.* **2017**, *11*, 585–596.
- Lee, W.K.; Ratnam, M.M.; Ahmad, Z.A. Detection of chipping in ceramic cutting inserts from workpiece profile during turning using fast Fourier transform (FFT) and continuous wavelet transform (CWT). *Precis. Eng.* **2017**, *47*, 406–423.
- Ni, S.H.; Li, J.L.; Yang, Y.Z.; Yang, Z.T. An improved approach to evaluating pile length using complex continuous wavelet transform analysis. *Insight Non-Destr. Test. Cond. Monit.* **2017**, *59*, 318–324.
- Rafiei, M.H.; Adeli, H. A novel unsupervised deep learning model for global and local health condition assessment of structures. *Eng. Struct.* **2018**, *156*, 598–607.
- Perez-Ramirez, C.A.; Amezquita-Sanchez, J.P.; Adeli, H.; Valtierra-Rodriguez, M.; Camarena-Martinez, D.; Romero-Troncoso, R.J. New methodology for modal parameters identification of smart civil structures using ambient vibrations and synchrosqueezed wavelet transform. *Eng. Appl. Artif. Intell.* **2016**, *48*, 1–12.
- Leigh, G.M. Fast FIR Algorithms for the Continuous Wavelet Transform From Constrained Least Squares. *IEEE Trans. Signal Process.* **2012**, *61*, 28–37.
- Unser, M. Fast Gabor-like windowed Fourier and continuous wavelet transforms. *IEEE Signal Process. Lett.* **1994**, *1*, 76–79.

17. Yi, H.; Chen, Z.; Cao, Y. High Precision Computation of Morlet Wavelet Transform for Multi-period Analysis of Climate Data. *J. Inf. Comput. Sci.* **2014**, *11*, 6369–6385.
18. Torrence, C.; Compo, G.P. A Practical Guide to Wavelet Analysis. *Bull. Am. Meteorol. Soc.* **1998**, *79*, 61–78.
19. Burrus, C.; Parks, T.W. *DFT/FFT and Convolution Algorithms: Theory and Implementation*; John Wiley & Sons Inc.: New York, NY, USA, 1991.
20. Jin, X.Q. *Preconditioning Techniques for Toeplitz Systems*; Higher Education Press: Beijing, China, 2010.
21. Nagy, J.G.; Plemmons, R.J.; Torgersen, T.C. Iterative image restoration using approximate inverse preconditioning. *IEEE Trans. Image Process.* **1996**, *5*, 1151–1162.
22. Proakis, J.G.; Manolakis, D.G. *Digital Signal Processing: Principles, Algorithms, And Applications*; Pearson Education: Delhi, India, 2007; Volume 23, pp. 392–394.
23. Mallat, S.G. *A Wavelet Tour of Signal Processing: The Sparse Way*; Academic Press: New York, NY, USA, 2009.
24. Yi, H.; Shu, H. The improvement of the Morlet wavelet for multi-period analysis of climate data. *Comptes Rendus Geosci.* **2012**, *344*, 483–497.
25. Jones, D.L.; Baraniuk, R.G. Efficient approximation of continuous wavelet transforms. *Electron. Lett.* **1991**, *27*, 748–750.
26. Zhao, Y.Y.; Yuan, X.; Wei, Y.H. Realization of Continuous Wavelet Transform of Sequences by MATLAB. *J. Sichuan Univ.* **2006**, *43*, 325–329.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).