New Wavelet Threshold Denoising Method in Noisy Blind Source Separation

Xuan-Sen He and Tian-Jiao Zhao

Abstract—In general conditions, most blind source separation algorithms are established on noisy-free model and ignore the noise that affects the quality of separated sources. Firstly, this paper introduces an improved natural gradient algorithm based on bias removal technology to estimate the demixing matrix under noisy environment. Then the discrete wavelet transform technology is applied to the separated signals to further remove noise. In order to improve the separation effect, this paper analyzes the deficiency of hard threshold and soft threshold, and proposes a new wavelet threshold function based on the wavelet decomposition and reconfiguration. The simulations have verified that this method improves the signal noise ratio (SNR) of the separation results and the separation precision.

Index Terms—Bias removal, blind source separation, gradient algorithm, wavelet threshold denoising.


1. Introduction

Blind source separation (BSS) is to recover the source signals only by using output signals of sensors when the source signals and mixing channel matrix are unknown. Generally, BSS algorithms are established on noisy-free models, such as natural gradient algorithm (NGA)\textsuperscript{[1]}, fast independent component analysis (ICA) algorithm\textsuperscript{[2][3]}, and equivariant adaptive separation for independence (EASI) algorithm\textsuperscript{[4]}. Those algorithms are the sub-optima in noisy environment. In order to solve the noisy BBS problems, a spontaneous idea is presented to remove the effect of noise, thus improving the algorithms.

The chief advantage of discrete wavelet transform (DWT) is the ability of partial and detailed analysis in different scales and the scale can be flexibly selected according to different applications. Because the main noisy energy is centralized in detailed component of wavelet decomposition, the wavelet threshold function can be applied to separated signals to remove noise.

In many practical situations, the observed signals can be modeled as

\[
x(t) = As(t) + n(t)
\]

where \( s(t) \in \mathbb{R}^m \) is an unknown source signals vector. The crucial assumption is that the source signals are statistically independent and the sources have zero mean value and unit variance. \( A \in \mathbb{R}^{m \times n} \) is an unknown mixing channel matrix, \( x(t) \in \mathbb{R}^n \) is an observed signals vector, and \( n(t) \in \mathbb{R}^n \) is the possible contaminating noise vector.

To recover the source signals given only the observed mixtures, the source signals can be recovered by

\[
y(t) = Wx(t) = \hat{s}(t)
\]

where \( W \in \mathbb{R}^{m \times m} \) is a separation matrix, \( y(t) \in \mathbb{R}^n \) is an estimate vector of \( s(t) \) using \( \hat{s}(t) \) denoting the estimate of \( s(t) \). In this paper, we assume that the number of source signals matches the number of mixtures, i.e., \( m=n \).

The NGA algorithm ignores the effect of noise, so the observed mixtures model is \( x(t) = As(t) \), and the updating equation of the separation matrix\textsuperscript{[1]} is

\[
W(t+1) = W(t) - \eta(t) \frac{\partial J(t)}{\partial W(t)}
= W(t) + \eta(t) \left[ I - \phi(y(t)) y^T(t) \right] W(t)
\]

where \( \eta(t) \) is a positive parameter known generally as the step size, \( J(t) \) is a cost function from which NGA is derived, \( \phi(y)=[\phi_1(y_1), \phi_2(y_2), \cdots, \phi_l(y_l)]^T \) is an vector of the nonlinear activation functions, and each \( \phi_l(y_l) \) depends on the source signal \( s_l(t) \) extracted at the \( l \)th output. Our aim is to reduce the bias in the estimated matrix \( W(t) \) of (3).

This paper is organized as follows. Section 2 is dedicated to deriving an improved NGA algorithm based on bias removal technology. In Section 3, a new wavelet threshold function is applied to separated signals to remove noise. Section 4 is the simulation results and Section 5 concludes the paper.
2. Improved NGA Algorithm

For simplicity in this derivation, it is assumed that \( n(t) \) is a zero-mean jointly Gaussian random vector, and the covariance matrix is \( R_{n,n} = E[n(t)n(t)'] \) whose \((i,j)\)th element is \( r_{ij} = E[n_i(t)n_j(t)] \), where \( E(x) \) is the expectation of \( x \). An unbiased estimate of \( \hat{W}(t) \) is obtained by the model\(^{5,6}\)

\[
\hat{W}(t + 1) = W(t) + \eta(t) \left[ I - \phi(\hat{y}(t)) \hat{y}(t)' \right] W(t) \tag{4}
\]

where \( \hat{y}(t) = W(t) A \hat{s}(t) \) is the estimate of \( y(t) \). Since \( \phi(y(t)) y'(t) \) in (3) is biased due to noise, we define the bias matrix to be \( M(t) \). If the noise is small, then \( \bar{E}[\phi(y(t)) y'(t)] = E[M(t)] \approx E[\phi(\hat{y}(t))] \hat{y}'(t) \) \( \hat{y}(t) \) can be written as \( y(t) = \hat{y}(t) + W(t)n(t) \), thus

\[
\bar{E}[\phi(y(t)) y'(t)] = E[\phi(\hat{y}(t)) + M(t)n(t)] \left( \hat{y}'(t) + M(t)n(t)' \right) = E[\phi(\hat{y}(t)) + W(t)n(t)] \hat{y}'(t) + W(t)n(t)' W(t) \] \tag{5}

So the \((i,j)\)th element of (5) is

\[
E(\hat{\phi}(y_{i,j}(t)) y_{i,j}'(t)) = E\left\{ \hat{\phi}(\hat{y}_{i,j}(t) + \sum_{l=1}^{n} w_{i,j}(l) n(t)) \hat{y}_{i,j}'(t) \right\} + E\left\{ \hat{\phi}(\hat{y}_{i,j}(t) + \sum_{l=1}^{n} w_{i,j}(l) n(t)) \sum_{p=1}^{\hat{n}(t)} w_{i,j}(p) n(p) \right\} \tag{6}
\]

Using theorem I in appendix A of [7], the second term on the right-hand side of (6) can be simplified as

\[
E\left\{ \hat{\phi}(\hat{y}_{i,j}(t) + \sum_{l=1}^{n} w_{i,j}(l) n(t)) \sum_{p=1}^{\hat{n}(t)} w_{i,j}(p) n(p) \right\} = E\left[ \frac{d\phi(\hat{y}(t))}{dy_{i,j}(t)} \sum_{l=1}^{n} w_{i,j}(l) n(t) \sum_{p=1}^{\hat{n}(t)} w_{i,j}(p) n(p) \right] \tag{7}
\]

where \( \frac{d\phi(\hat{y}(t))}{dy_{i,j}(t)} \) is the derivative of \( \phi(\hat{y}(t)) \). A Taylor series expansion of \( \phi(y_{i,j}(t)) \) in the first term gives

\[
\phi(y_{i,j}(t)) = \phi(\hat{y}_{i,j}(t) + \sum_{l=1}^{n} w_{i,j}(l) n(t)) = \phi(\hat{y}_{i,j}(t)) + \frac{d\phi(\hat{y}_{i,j}(t))}{dy_{i,j}(t)} \sum_{l=1}^{n} w_{i,j}(l) n(t)
\]

\[
= \phi(\hat{y}_{i,j}(t)) + \frac{1}{2} \frac{d^2\phi(\hat{y}_{i,j}(t))}{dy_{i,j}(t)^2} \left[ \sum_{l=1}^{n} w_{i,j}(l) n(t) \right]^2 + \sum_{l=1}^{n} \sum_{p=1}^{\hat{n}(t)} \sum_{j=1}^{\hat{n}(t)} n_{i,j}(n_{i,p} n_{j,q}(t)) \tag{8}
\]

where \( \frac{d^2\phi(\hat{y}(t))}{dy_{i,j}(t)^2} \) is the second order derivative of \( \phi(\hat{y}(t)) \).

Because the noise is assumed to be small and \( o(n(t)n(t)) \) is a higher order infinitesimal, we take expectations of both sides of (8) and neglect the terms in \( o(n(t)n(t)) \). Combining (5)-(8), the first term on the right-hand side of (6) is simplified as

\[
\bar{E}[\phi(y(t)) y'(t)] = E[\phi(\hat{y}(t)) y'(t)] + E\left[ \frac{d\phi(\hat{y}(t))}{dy} \hat{y}'(t) W(t)n(t) \right] + \frac{1}{2} \text{diag} \left[ W(t) R_{n,n} W(t) \right] E\left[ \frac{d^2\phi(\hat{y}(t))}{d^2y} \hat{y}'(t) \right] \tag{9}
\]

Defining

\[
d^2\phi(y) = \left[ \frac{d^2\phi(y_{i,j}(t))}{dy_{i,j}(t)^2}, \ldots, \frac{d^2\phi(y_{i,j}(t))}{dy_{i,j}(t)^2} \right]' \]

and substituting (7)-(9) into (5), we see that

\[
\bar{E}[\phi(y(t)) y'(t)] = E[\phi(\hat{y}(t)) y'(t)] + E[\Phi_s(y(t))] W(t) R_{n,n} W(t)
\]

\[
+ \frac{1}{2} \text{diag} \left[ W(t) R_{n,n} W(t) \right] E\left[ \frac{d^2\phi(\hat{y}(t))}{d^2y} \hat{y}'(t) \right] \tag{10}
\]

where the \((i,j)\)th elements of the diagonal matrix \( \text{diag}[\Phi_s(y(t))] \) and \( \text{diag}[W(t) R_{n,n} W(t)] \) are \( E\left[ \frac{d\phi(y_{i,j}(t))}{dy_{i,j}(t)} \right] \) and

\[
\sum_{l=1}^{\hat{n}(t)} \sum_{j=1}^{\hat{n}(t)} \sum_{p=1}^{\hat{n}(t)} r_{i,j} r_{i,p} w_{i,j}(t) \]

respectively. From (10) an instantaneous estimate of \( \phi(\hat{y}(t)) \hat{y}'(t) \) can be given as

\[
\phi(\hat{y}(t)) \hat{y}'(t) = \phi(y(t)) y'(t) - \Phi_s(y(t)) W(t) R_{n,n} W(t)
\]

\[
- \frac{1}{2} \text{diag} \left[ W(t) R_{n,n} W(t) \right] E\left[ \frac{d^2\phi(\hat{y}(t))}{d^2y} \hat{y}'(t) \right] \tag{11}
\]

The last term on the right-hand side of (11) depends on \( \hat{y}(t) \), but it only affects the scaling of the output signals at convergence, in which case \( \bar{E}[\phi(\hat{y}(t)) \hat{y}'(t)] \) and \( E\left[ \frac{d^2\phi(\hat{y}(t))}{d^2y} \hat{y}'(t) \right] \) are diagonal. So we can neglect the last term on the right-hand side of (11). Substituting the right-hand side of (11) into (4), we have

\[
W(t + 1) = W(t) + \eta(t) \left[ I - \phi(\hat{y}(t)) \hat{y}'(t) \right] + \Phi_s(y(t)) W(t) R_{n,n} W(t) \tag{12}
\]

In case \( R_{n,n} = \sigma_n^2 I \) and \( \sigma_n^2 = E[n_i(t)^2] \), where \( I \) is the identity matrix and \( \sigma_n^2 \) is the variance of \( n(t) \), Equation (12) becomes
\[ W(t+1) = W(t) + \eta(t) \left[ I - \Phi(\hat{y}(t))\hat{y}(t) \right] + \sigma^2_n \Phi(\hat{y}(t)) W(t) W^T(t) W(t). \]  

(13)

If the elements of \( n(t) \) are not Gaussian-distributed, (12) and (13) are still appropriate as long as the power in each element of \( n(t) \) is small.

3. Wavelet Threshold Denoising

In noisy blind source separation, the estimated signals vector is

\[ \hat{y} = Wx = \hat{s} + Wn = \hat{s} + n, \]  

(14)

where \( \hat{y} \) is the estimate vector of the source signals containing noise. In order to recover the source signals accurately, we must remove noise from separated signals.

After the source signals are wavelet transformed, the coefficients of signals have biggish values only in certain scale and position, but the coefficients of noise have similar values in all scales and positions. Through estimating the amplitude value of noise in every scale and comparing the amplitude value of signal, we can remove the wavelet coefficients of noise so as to remove noise by setting threshold function. The above-mentioned idea is the foundation of the wavelet threshold denoising.

3.1 Processing Steps

Step 1: DWT. For \( j = 0, 1, \cdots, J; k = 0, 1, \cdots, N \), the DWT of (14) is

\[ \omega_{j,k} = \omega_j(k) + \omega_n(j,k) \]  

(15)

where \( \omega_j(j,k) \), \( \omega_n(j,k) \), and \( \omega_n(j,k) \) are the wavelet coefficients on \( j \)th layer of noisy signal, source signal, and noise, respectively. \( J \) is the maximal number of the wavelet decomposition levels, and \( N \) is the length of signal.

Step 2: establishing the threshold function. The high frequency wavelet coefficients are quantitatively treated from the 1st to \( N \)th level through selecting appropriate threshold function.

Step 3: reconstructing the source signals. By using the characteristic of the low frequency coefficients on the \( N \)th level and the high frequency coefficients from the 1st to \( N \)th level in DWT domain, the wavelet reconstruction can be completed.

The key step is Step 2, that is, how to select and quantify the threshold function has a close bearing on the quality of denoising.

3.2 Selection of the Threshold Function

The most commonly used threshold function is the fixed threshold\(^{[9]} \) and the selecting methods include two kinds: hard and soft threshold functions. The hard threshold function is

\[ \hat{\omega}_{j,k} = \begin{cases} \omega_{j,k} > \lambda & \Rightarrow & \hat{\omega}_{j,k} = 0 \\ \omega_{j,k} < \lambda & \Rightarrow & \hat{\omega}_{j,k} = \omega_{j,k} \end{cases} \]  

(16)

and the soft threshold function is

\[ \hat{\omega}_{j,k} = \begin{cases} \text{sgn}(\omega_{j,k})(|\omega_{j,k}| - \lambda) & \Rightarrow & |\omega_{j,k}| \geq \lambda \\ 0 & \Rightarrow & |\omega_{j,k}| < \lambda \end{cases} \]  

(17)

where \( \lambda = \sigma_s \sqrt{2 \log(N)} \) is the threshold, \( \sigma_s \) is the standard deviation of wavelet coefficient of noise, and \( \text{sgn}(\omega) \) is the sign function.

In many cases, such as on the discontinuous points, the Gibbs phenomenon will occur, when we use the hard threshold function. If we use the soft threshold method, there is constant deviation between estimation and decomposition wavelet coefficients, and it can not represent the distribution of the signal energy.

To overcome the above-mentioned limitations, many new threshold functions are put forward, such as the incorporation of soft and hard thresholds\(^{[9]} \).

\[ \hat{\omega}_{j,k} = \begin{cases} \text{sgn}(\omega_{j,k})(|\omega_{j,k}| - \alpha \lambda) & \Rightarrow & |\omega_{j,k}| \geq \lambda \\ 0 & \Rightarrow & |\omega_{j,k}| < \lambda \end{cases} \]  

(18)

where \( 0 < \alpha \leq 1 \). When \( \alpha = 0 \), the threshold function is equivalent to the hard threshold function. When \( \alpha = 1 \), the threshold function is equivalent to the soft threshold function, but this method does not have the smoothing action. Another one is the modulus square processing method\(^{[10]} \).

\[ \hat{\omega}_{j,k} = \begin{cases} \text{sgn}(\omega_{j,k})(|\omega_{j,k}| - \alpha \lambda^2) & \Rightarrow & |\omega_{j,k}| \geq \lambda \\ 0 & \Rightarrow & |\omega_{j,k}| < \lambda \end{cases} \]  

(19)

where \( 0 < \alpha \leq 1 \). When \( |\omega_{j,k}| \geq \lambda \), \( \hat{\omega}_{j,k} \) is a nonlinear function. When \( |\omega_{j,k}| \) keeps on increasing, the method is close to the hard threshold function, but it still loses the high frequency information. Besides, the threshold median filter denoising method\(^{[11]} \) is also adopted. However, the threshold functions of (18) and (19) also have disadvantages when the noise is not Gaussian distribution.

In order to improve the performance of denoising, through a great deal of experiments, a new threshold function is presented in this paper as follows:

\[ \hat{\omega}_{j,k} = \begin{cases} 0 & \Rightarrow & |\omega_{j,k}| \leq \lambda_1 \\ \text{sgn}(\omega_{j,k}) \frac{\lambda_2^2 - |\omega_{j,k}|^2}{\lambda_2 - \lambda_1} & \Rightarrow & \lambda_1 < |\omega_{j,k}| \leq \lambda_2 \\ \omega_{j,k} & \Rightarrow & |\omega_{j,k}| > \lambda_2 \end{cases} \]  

(20)

In many cases, such as on the discontinuous points, the Gibbs phenomenon will occur, when we use the hard threshold function. If we use the soft threshold method, there is constant deviation between estimation and decomposition wavelet coefficients, and it can not represent the distribution of the signal energy.
where \( \lambda_2 = \lambda = \sigma_\epsilon \sqrt{2 \log(N)} \), \( \lambda_1 \in [0, \lambda_2] \). In case noise is Gaussian distribution, selecting \( \lambda_1 = 2 \lambda_2/3 \), the denoising effect is very good. If the noise is not Gaussian distribution, through selecting the proportional relations between \( \lambda_1 \) and \( \lambda_2 \), we can get an ideal result.

To verify the performance of the threshold function, we compare the estimation effect of different threshold functions. The result is shown in Fig. 1. From Fig. 1, it can be seen that the threshold function of (20) is divided into three stages, so this method can more effectively remove the noise compared with other threshold functions.

### 4. Simulation

In simulation, the source signals \( s(t) \) are respectively: sign function \( s_1(t) = \text{sgn}(\cos(2\pi t 155)) \), low frequency sine signal \( s_2(t) = \sin(2\pi t 90) \), and amplitude modulation signal \( s_3(t) = \sin(2\pi t 9) \sin(2\pi t 300) \). The signals are sampled at 10 kHz and the length of data samples is 4000. The waveform of the source signals is shown in Fig. 2. The mixing matrix \( A \sim U(-1, 1) \), i.e. the elements of \( A \) are uniformly distributed in \([-1, 1]\). In this experiment, non-Gaussian and Gaussian types of noise are used to test the method.

1) The noise is Gaussian distribution with zero mean and variance 0.01. The observed signal waveform is shown in Fig. 3.

According to (12) the separation matrix \( W \) is updated, where nonlinear activation function \( \phi(y) = y^3 \), step-size \( \eta(t) = 0.005 \), and the separated signals waveform is shown in Fig. 4. Then the separated signals are decomposed by db3 wavelet packet at three levels and reconstructed by wavelet packet. The wavelet denoising signals waveform is shown in Fig. 5.
Fig. 6. Waveform of the cross-talking error.

Fig. 7. PI waveform.

To test the performance of the proposed method, we use the cross-talking error PI:

$$PI = \sum_{i=1}^{n} \left[ \sum_{j=1}^{n} \frac{c_{i,j}}{\max_j c_{i,j}} - 1 \right] + \sum_{j=1}^{n} \left[ \sum_{i=1}^{n} \frac{c_{i,j}}{\max_i c_{i,j}} - 1 \right]$$

(21)

where $\mathbf{C} = W \mathbf{A}$ is the system matrix, and $c_{i,j}$ is the $(i, j)$th element of the matrix $\mathbf{C}$. The PI is shown in Fig. 6. From Fig. 6, it can be seen that the proposed method is feasible and the signals have been successfully separated when iteration times are about 1500 and $PI=0.4425$.

2) The noise is sub-Gaussian distribution with variance 0.01, and the noise is uniformly distributed in $[-1, +1]$. PI is used to verify the proposed method. When iteration times are about 1000, the signals have been successfully separated and $PI=0.3397$. The result is shown in Fig. 7.

The effect of denoising can be measured by signal noise ratio (SNR)

$$SNR(y_i) = 10 \log \frac{E[y_i^2]}{E[(y_i - s_i)^2]}$$

(22)

We can find that wavelet denoising method gives more accurate estimation of sources by contrasting Fig. 4 with Fig. 5. After repeated experiments, the proposed method can improve the SNR about 1 dB. The SNR comparison of different threshold functions and different algorithms is shown in Table 1 and Table 2. Table 1 shows that the denoising effect of the proposed method is the best. From Table 2 it can be seen that the SNR of the improved NGA is increased but the effects are not visible. The proposed method obviously improves the SNR compared with other algorithms.

### Table 1: Comparison of different threshold functions

<table>
<thead>
<tr>
<th>Threshold function</th>
<th>SNR(dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard threshold function</td>
<td>-2.1969</td>
</tr>
<tr>
<td>Soft threshold function</td>
<td>-2.1969</td>
</tr>
<tr>
<td>Incorporation of soft and hard threshold function</td>
<td>-2.1694</td>
</tr>
<tr>
<td>Modulus square threshold function</td>
<td>-2.1655</td>
</tr>
<tr>
<td>Proposed threshold function</td>
<td>-1.4086</td>
</tr>
</tbody>
</table>

### Table 2: Comparison of different algorithms

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>SNR(dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NGA</td>
<td>-2.6445</td>
</tr>
<tr>
<td>Improved NGA</td>
<td>-2.4895</td>
</tr>
<tr>
<td>NGA+threshold function</td>
<td>-1.1021</td>
</tr>
<tr>
<td>Proposed method</td>
<td>-1.0385</td>
</tr>
</tbody>
</table>

5. Conclusions

This paper introduces a new improved NGA based on bias removal technology and the DWT combined with a novel threshold function to remove the noise of the separated signals. Then we can separate more precise source signals. The simulation results prove that the proposed method has a good separating efficiency, higher anti-noise ability, and SNR. But the method is produced on the premise that the noise power is known, and the less the noise power is, the better the separation effect is. This is the limitation of the method. How to implement the noisy signals blind separation with big noise power is our next research work.

### References


**Xuan-Sen He** was born in Henan Province, China, in 1958. He received the B.S. degree in electrical engineering in 1983 and the M.S. degree in signal and information processing in 1995, both from National University of Defense Technology (NUDT), Changsha. He has worked with the College of Computer and Communication, Hunan University since 2002. His research interests include stochastic signal processing, blind source separation, digital signal processing, and information security technology.

**Tian-Jiao Zhao** was born in Gansu Province, China, in 1985. She received the B.S. degree in electrical information engineering from Qingdao University of Science and Technology in 2007. She is currently pursuing her M.S. degree with the College of Computer and Communication, Hunan University. Her research interests include blind source separation and digital signal processing.