ỨNG DỤNG PHƯƠNG PHÁP WAVELET TRONG KHỬ NHIỄU CHUỖI THỜI GIAN

Tô Anh Dũng, Hoàng Văn Hà
Trường Đại học Khoa học Tự nhiên, ĐHQG-HCM
(Bài nhận ngày 22 tháng 08 năm 2007, hoàn chỉnh sửa chữa ngày 29 tháng 09 năm 2007)

TÓM TẮT: *Bài này giới thiệu về phương pháp phân tích wavelet, so sánh một số điểm của phương pháp này với phép phân tích Fourier. Trên cơ sở đó trình bày phép biến đổi wavelet rời rạc để khử nhiễu chuỗi thời gian rời rạc, cách khử nhiễu này dựa trên cách chọn hàm wavelet, ước lượng phương sai nhiễu, xác định ngưỡng. Cuối cùng đưa ra một số độ đo để so sánh sai số và tính hiệu quả của các cách khử nhiễu khác nhau.*

1. ĐẶT VẤN ĐỀ

Trong bài toán khử nhiễu chuỗi thời gian trước đây phương pháp phân tích Fourier thường được sử dụng. Tuy nhiên các hàm Fourier của phép phân tích Fourier chỉ sử dụng một tham số là tần số, điều này sẽ làm mất đi các thông tin về thời gian dẫn đến việc tính toán khó khăn. Mặt khác, biến đổi Fourier lại kém thích hợp đối với các chuỗi thời gian không trơn và có đỉnh nhọn, nhưng hàm wavelet lại phân tích rất tốt dữ liệu dạng này. Do đó biến đổi wavelet đã tỏ ra vượt trội và khắc phục được các nhược điểm của phương pháp Fourier.

Bài báo này khảo sát mức độ hiệu quả của khử nhiễu chuỗi thời gian với phương pháp wavelet trên số liệu mẫu.

2. GIỚI THIỆU VỀ WAVELET

2.1 Định nghĩa

Wavelet là một họ các hàm số có tính chất địa phương hóa theo thời gian hoặc không gian. Ta thu được chúng từ một hàm đơn $\psi(x)$, gọi là hàm wavelet mẹ, bằng các phép tịnh tiến và co giãn. Hàm wavelet phải thỏa các điều kiện sau đây:

$$\int_{-\infty}^{+\infty} \psi(x)\,dx = 0 \qquad (1.1)$$

$$\int_{-\infty}^{+\infty} \psi^2(x)\,dx = 1 \qquad (1.2)$$

$$c_\psi = \int_0^{+\infty} \frac{|\Psi(f)|^2}{f} df \quad \text{thỏa} \quad 0 < C_\psi < \infty \qquad (1.3)$$

với Ψ là biến đổi Fourier của ψ:

$$\Psi(f) = \int_{-\infty}^{+\infty} \psi(x) \cdot e^{-i2\pi fx} dx$$

Với một hàm wavelet mẹ cho trước, $\forall\ a, b\ (a \neq 0)$, ta xây dựng được họ các hàm wavelet bằng phép tịnh tiến và cõ giãn từ $\psi(x)$ như sau:

$$\psi_{a,b}(x) = |a|^{-1/2} \psi\left(\frac{x-b}{a}\right) \qquad (1.4)$$

2.2 Ví dụ

Một số các hàm wavelet mẹ thường dùng:
Hàm nón Mexico:

$$\psi^{(MH)}(x) = (1-x^2)e^{-x^2/2},\ -\infty < x < +\infty$$

Hàm Haar:

$$\psi^{(H)}(x) = \begin{cases} 1, & 0 \le x < 1/2 \\ -1, & 1/2 < x \le 1 \\ 0, & \text{nơi khác} \end{cases}$$

3. BIẾN ĐỔI WAVELET RỜI RẠC

3.1 Giới thiệu

Trong phần 2 này, ta sẽ trình bày biến đổi wavelet rời rạc đối với một chuỗi thời gian rời rạc ([1], [2], [3]).

Chuỗi thời gian $\{X_t, t = 0,1,...,N-1\}$ có chiều dài là $N = 2^J$ với J là một số nguyên dương, sau khi qua biến đổi wavelet rời rạc các giá trị X_t sẽ được chuyển thành các hệ số wavelet theo phương trình sau:

$$\mathbf{W} = \mathbf{M}\,\mathbf{X} \tag{2.1}$$

Với $\mathbf{W}$ là ma trận chứa các hệ số của biến đổi wavelet rời rạc gọi là hệ số wavelet, $\mathbf{M}$ là ma trận trực giao được xây dựng t

3.2 Lọc wavelet và lọc co giãn

Dãy $\{h_l,\ l = 0,...,L-1\}$ có bề rộng là số chẵn L thỏa các điều kiện sau thì được gọi là lọc wavelet:

$$\sum_{l=0}^{L-1} h_l = 0 \tag{2.2}$$

$$\sum_{l=0}^{L-1} h_l^2 = 1 \tag{2.3}$$

$$\sum_{l=0}^{L-1} h_l h_{l+2n} = \sum_{l=-\infty}^{+\infty} h_l h_{l+2n} = 0,\ \forall n \in \square \tag{2.4}$$

Từ điều kiện (2.3) và (2.4) ta suy ra được tính chất trực giao của lọc wavelet. Tương tự như lọc wavelet, lọc co giãn là dãy $\{g_l,\ l = 0,...,L-1\}$ thỏa điều kiện

$$\sum_{l=0}^{L-1} g_l = \sqrt{2}$$

và các điều kiện (2.3), (2.4). Như vậy cả lọc wavelet và lọc co giãn đều thỏa tính chất trực giao và có mối liên hệ như sau

$$h_l = (-1)^l g_{L-l-1} \text{ vaø } g_l = (-1)^{l+1} h_{L-l-1},\ l = 0,...,L-1$$

Bây giờ ta sẽ đi vào thuật toán chính của biến đổi wavelet rời rạc–thuật toán kim tự tháp.

3.3 Thuật toán kim tự tháp

Nếu xét chuỗi thời gian $\{X_t, t = 0,1,...,N-1\}$ có chiều dài là $N = 2^J$, thì thuật toán kim tự tháp bao gồm J bước. Ta đặt $\{V_{0,t} = X_t, t = 0,...,N-1\}$ là chuỗi được xử lý ở bước một thì ở bước thứ j chuỗi được xử lý kế tiếp sẽ là $\{V_{j-1,t},\ t = 0,...,N_{j-1}\}$ với $N_{j-1} = \frac{N}{2^j}$. Ở dạng ma trận, sau bước thứ nhất chuỗi $\mathbf{X} = \mathbf{V}_0$ sẽ được chia thành hai vectơ $\mathbf{V}_1$ và $\mathbf{W}_1$ mỗi vectơ có chiều dài là $N/2$ chứa các hệ số co giãn và hệ số wavelet, tương tự ở bước thứ hai thì vectơ $\mathbf{V}_1$ được xem như chuỗi thời gian ban đầu và được chia thành hai vectơ $\mathbf{V}_2$ và $\mathbf{W}_2$ có chiều dài là $N/4$, lập lại như vậy sau J bước ta thu được các vectơ $\mathbf{W}_1, \mathbf{W}_2, ..., \mathbf{W}_J, \mathbf{V}_J$ với hai vectơ sau cùng mỗi vectơ chỉ chứa một phần tử tạo thành ma trận $\mathbf{W}$ như sau

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \vdots \\ \mathbf{W}_J \\ \mathbf{V}_J \end{bmatrix} \tag{2.5}$$

là ma trận chứa các hệ số wavelet.

Các hệ số $W_{j,t}$ và $V_{j,t}$ trong các vectơ $\mathbf{W}_j$, $\mathbf{V}_j$ được tính bởi các biểu thức

$$W_{j,t} = \sum_{l=0}^{L-1} h_l V_{j-1,2t+1-l \bmod N_{j-1}}$$

và

$$V_{j,t} = \sum_{l=0}^{L-1} g_l V_{j-1,2t+1-l \bmod N_{j-1}}$$

$$t = 0,...,N_{j-1}.$$

Do các ma trận $\mathbf{M}$ và $\mathbf{W}$ có tính trực giao, nên từ (2.1) với ký hiệu $\mathbf{A}'$ là chuyển vị của $\mathbf{A}$, ta nhận được

$$\mathbf{X}=\mathbf{MW}=\sum_{j=1}^{J}\mathbf{M}_j\mathbf{W}_j+\mathbf{\Lambda}_J\mathbf{V}_J=\sum_{j=1}^{J}\mathbf{D}_j+\mathbf{S}_j \quad (2.6)$$

(2.6) được gọi là phân tích đa phân giải trong biến đổi wavelet rời rạc của chuỗi thời gian.

4. KHỬ NHIỄU TRONG CHUỖI THỜI GIAN

4.1 Giới thiệu

Trong phần 4 này, ta ứng dụng phép biến đổi wavelet rời rạc đã trình bày ở phần trên để khử nhiễu một chuỗi thời gian, chuyển các giá trị X_t của chuỗi thời gian $\mathbf{X}$ thành các hệ số wavelet, sau đó dùng hàm ngưỡng để khử đi các nhiễu.

Giả sử chuỗi thời gian quan sát được $\{X_t, t: 0,...,N-1\}$ chứa nhiễu có dạng

$$\mathbf{X}=\mathbf{D}+\boldsymbol{\varepsilon} \quad (3.1)$$

với,

$\mathbf{X}=(X_0,X_1,...,X_{N-1})$: vectơ chuỗi thời gian quan sát được,

$\mathbf{D}=(D_0,D_1,...,D_{N-1})$: chuỗi không bị nhiễu cần tìm,

$\boldsymbol{\varepsilon}=(\varepsilon_0,\varepsilon_1,...,\varepsilon_{N-1})$: vectơ nhiễu.

Chúng ta sẽ xét bài toán với các giả thiết

- Chiều dài *N* của chuỗi thời gian $\mathbf{X}$ bằng 2^J với *J* nguyên dương.
- $\mathbf{X}$ là chuỗi thời gian dừng.
- Nhiễu $\boldsymbol{\varepsilon}$ có dạng ồn trắng.

Phương pháp khử nhiễu gồm ba bước:

1. Sử dụng biến đổi wavelet rời rạc để đưa $\mathbf{X}$ về ma trận các hệ số wavelet $\mathbf{W}$:

$$\mathbf{W}=\mathbf{MX}=\mathbf{MD}+\mathbf{M}\boldsymbol{\varepsilon}$$

2. Hiệu chỉnh các hệ số wavelet bằng hàm ngưỡng, thu được ma trận ký hiệu là $\hat{\mathbf{W}}$:

$$\mathbf{W}\Rightarrow\hat{\mathbf{W}}$$

3. Dùng biến đổi wavelet ngược để thu lại ước lượng đã khử nhiễu:

$$\hat{\mathbf{X}}=\mathbf{W}^{-1}\hat{\mathbf{W}}$$

Để khử nhiễu được hiệu quả, ngoài việc chọn hàm wavelet thích hợp còn phụ thuộc vào ba yếu tố sau:

- Ước lượng phương sai của nhiễu $\hat{\sigma}_\varepsilon$.
- Chọn hàm ngưỡng $\delta_T(.)$.
- Xác định ngưỡng *T*.

4.2 Ước lượng nhiễu

Từ phương trình (3.1) với giả thiết nhiễu $\boldsymbol{\varepsilon}$ có dạng ồn trắng $\varepsilon_t \sim N(0,\sigma^2), \forall t$. Để ước lượng phương sai σ^2 ta dùng phương pháp gọi là độ lệch trung vị tuyệt đối (MAD) được xét trong [2], ta ước lượng dựa trên *N/2* hệ số wavelet trong $\mathbf{W}_1$ thu được từ bước thứ nhất của biến đổi wavelet rời rạc:

$$\hat{\sigma}_{MAD}=\frac{\text{median}\left\{\left|W_{1,0}\right|,\left|W_{1,1}\right|,...,\left|W_{1,\frac{N}{2}-1}\right|\right\}}{0,6745} \quad (3.2)$$

4.3 Hàm ngưỡng

Cách chọn hàm ngưỡng khác nhau để hiệu chỉnh các hệ số wavelet sẽ dẫn đến sự khác biệt trong khử nhiễu, sau đây là các loại hàm ngưỡng thường dùng:

Ngưỡng cứng:

$$\delta_T^H(W_{i,j})=W\mathbf{1}_{\{|W_{i,j}|>T\}} \quad (3.3)$$

Ngưỡng mềm:

$$\delta_T^H(W_{i,j})=\text{sgn}(W_{i,j})(|W_{i,j}|-T)\mathbf{1}_{\{|W_{i,j}|>T\}} \quad (3.4)$$

với $\mathbf{1}_{\{|W_{i,j}|>T\}}$ là hàm chỉ tiêu, $\text{sgn}(W_{i,j})$ là dấu của $W_{i,j}$.

4.4. Xác định ngưỡng T

Ở nghiên cứu này chúng tôi dùng ngưỡng phổ dụng ([1], [2]). Ngưỡng này không phụ thuộc vào cách xác định hàm ngưỡng mà chỉ phụ thuộc vào phương sai σ^2 của nhiễu. Với $\mathbf{X}$ là chuỗi thời gian có chiều dài là N, thì ngưỡng phổ dụng được xác định là:

$$T = \sigma\sqrt{2\log N} \tag{3.5}$$

Một loại ngưỡng nữa được sử dụng trong bài này là ngưỡng SURE (Ước lượng mạo hiểm không chệch Stein) ([1], [2]). Gọi $\mathbf{W}=\{W_{i,j}\}$ là ma trận các hệ số wavelet, thì ngưỡng SURE được tính dựa trên ước lượng:

$$E\left\|\hat{\mathbf{M}}^{(T)} - \mathbf{W}\right\|^2 \tag{3.6}$$

Khi đó T được xác định để (3.6) đạt cực tiểu:

$$SURE(T,\mathbf{W}) = N - 2.\#\left\{ W_{j,k} : \left|W_{j,k}\right| \le T \right\} + \sum_j \sum_k \left(\min\left|W_{j,k}\right|, T\right)^2$$

Trong đó: $\#\left\{W_{j,k} : \left|W_{j,k}\right| \le T\right\}$ là số các phần tử $W_{j,k}$ thỏa $\left|W_{j,k}\right| \le T$ và $\hat{\mathbf{W}}^{(T)}$ là ma trận các hệ số wavelet được hiệu chỉnh bằng ngưỡng T.

Ngoài ra còn nhiều phương pháp khác để xác định ngưỡng T như kiểm định giả thiết, phương pháp Bayes, minimax… mà chúng tôi sẽ khảo sát trong các bài sau.

5. SO SÁNH SAI SỐ

5.1 Định nghĩa

Sau khi khử nhiễu chuỗi thời gian, vấn đề đặt ra là làm sao biết được hiệu quả của việc khử nhiễu? Chuỗi thời gian đã khử được bao nhiêu phần trăm nhiễu ? Như vậy ta cần lập ra một độ đo để so sánh chuỗi thời gian trước và sau khi khử nhiễu.

Gọi $\mathbf{X}^o = \left\{X_i^o\right\}_{i=\overline{1, N-1}}$, $\mathbf{X}^{noise} = \left\{X_i^{noise}\right\}_{i=\overline{1, N-1}}$, $\mathbf{X}^{den} = \left\{X_i^{den}\right\}_{i=\overline{1, N-1}}$ lần lượt là các vectơ chuỗi thời gian ban đầu (chưa bị nhiễu), bị nhiễu và sau khi khử nhiễu. Vì độ dài các chuỗi là N nên với mỗi $i \in \{0, 1, \ldots, N\text{-}1\}$, khoảng cách $\left|X_i^{den} - X_i^o\right|$ sẽ biểu diễn một sai số của từng điểm thuộc chuỗi thời gian sau khi khử nhiễu so với chuỗi gốc. Và khoảng cách

$$\left\|\mathbf{X}^{den} - \mathbf{X}^o\right\| = \sum_{i=0}^{N-1}\left|X_i^{den} - X_i^o\right| \tag{4.1}$$

biểu thị sai số tích lũy cho hai chuỗi. Tuy nhiên (4.1) không cho biết được chuỗi $\mathbf{X}^o$ được khử nhiễu hoàn toàn đến bao nhiêu. Chúng tôi đưa ra một công thức khác gọi là sai số trung bình

$$ME = \frac{1}{N}\sum_{i=0}^{N-1}\left|\frac{X_i^{den} - X_i^o}{X_i^o}\right| \tag{4.2}$$

Về mặt ý nghĩa, (4.2) cho biết phần trăm sai số trung bình giữa chuỗi sau khi khử nhiễu và chuỗi gốc. Để so sánh hiệu quả khử nhiễu, tức là nhiễu được loại bỏ đi bao nhiêu sau khi áp dụng phương pháp khử nhiễu, chúng tôi so sánh tỷ lệ chênh lệch trung bình của sai số trước và sau khi khử nhiễu và đưa ra công thức sau:

$$MRE = \frac{1}{N}\sum_{i=0}^{N-1}\left|\frac{X_i^{den} - X_i^o}{X_i^{noise} - X_i^o}\right| \tag{4.3}$$

Với $\left|X_i^{noise} - X_i^o\right|$ và $\left|X_i^{den} - X_i^o\right|$ là sai số trước và sau khi khử nhiễu, ta gọi (4.3) là tỷ số sai số trung bình khử nhiễu. Công thức (4.3) biểu thị tỷ lệ của phần đã khử được và phần trước khi khử nhiễu, qua đó ta biết lượng nhiễu đã được khử bao nhiêu phần trăm.

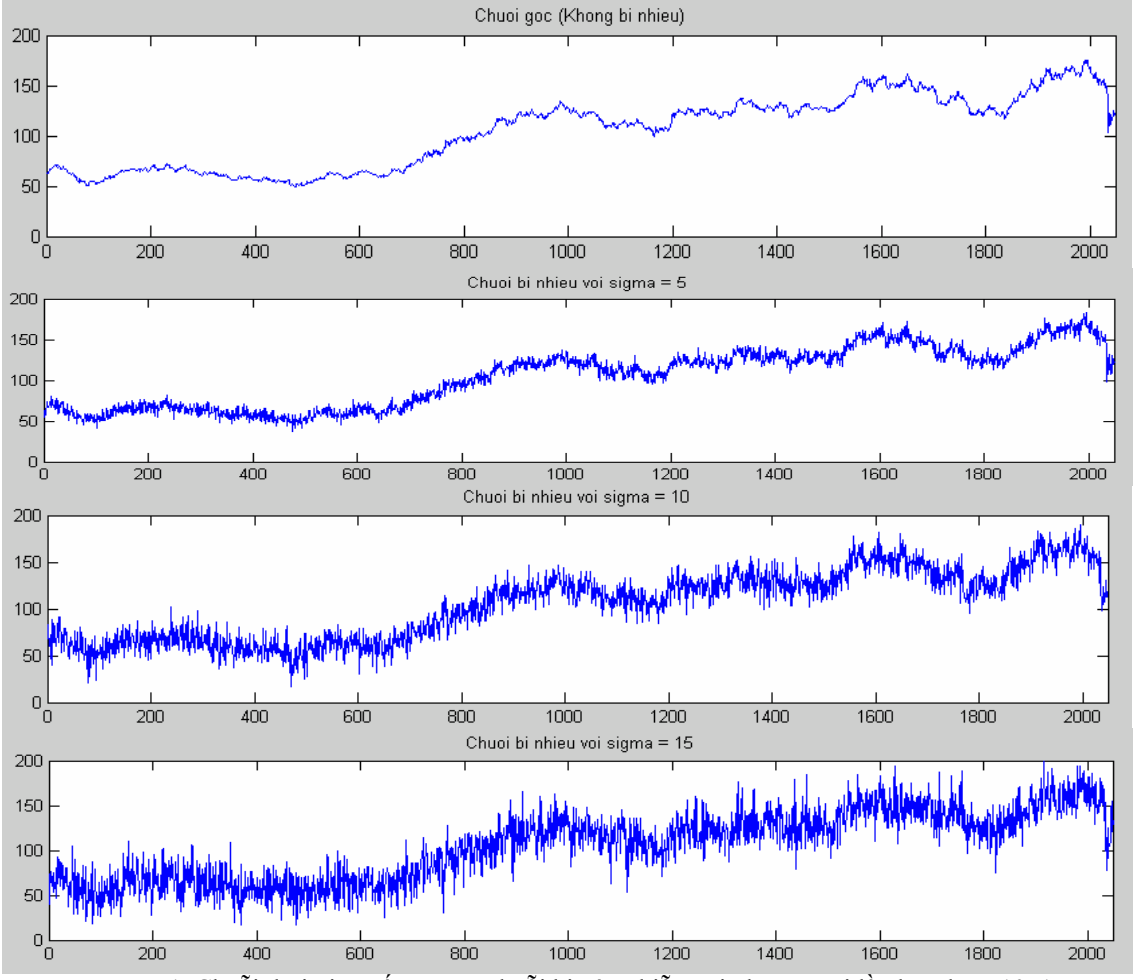
Ngoài ra, chúng tôi sử dụng sai số bình phương trung bình để đo độ lệch giữa hai chuỗi trước và sau khi khử nhiễu

$$MSE = \frac{1}{N}\left(\sum_{i=0}^{N-1}\left|\frac{X_i^{den} - X_i^o}{X_i^o}\right|^2\right)^{1/2} \quad (4.4)$$

5.2 So sánh sai số khử nhiễu chuỗi thời gian

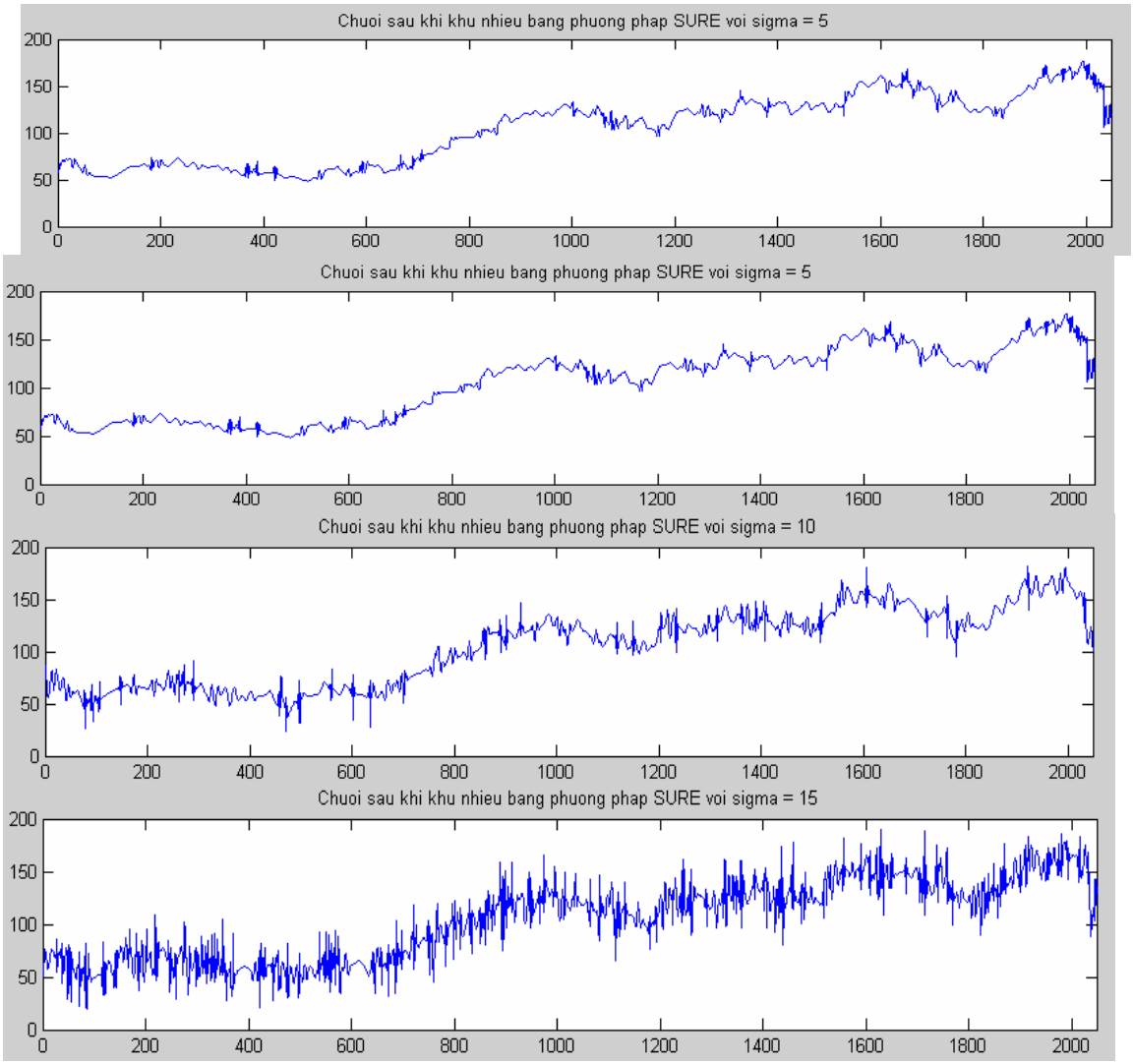
Ở mục này ta sẽ khử nhiễu một chuỗi thời gian cụ thể và tính các sai số. Chuỗi được sử dụng ở đây là dữ liệu về chỉ số chứng khoán hàng ngày của công ty IBM gồm 2048 điểm (Nguồn: Time Series Library). Chúng tôi sử dụng phần mềm Matlab và gói Wavelab để khử nhiễu chuỗi thời gian này. Để tiện so sánh, chúng tôi tạo ra ba phiên bản chuỗi thời gian khác nhau: (1)- chuỗi gốc chưa bị nhiễu (2)- chuỗi gốc bị gây nhiễu với các phương sai nhiễu lần lượt là σ^2= 5, 10, 15 và (3)- chuỗi sau khi khử nhiễu dùng hai ngưỡng là SURE và phổ dụng.

Sau khi gây nhiễu chuỗi gốc với ba phương sai khác nhau, các chuỗi mới chứa nhiễu có đồ thị như ở hình 1 dưới đây.



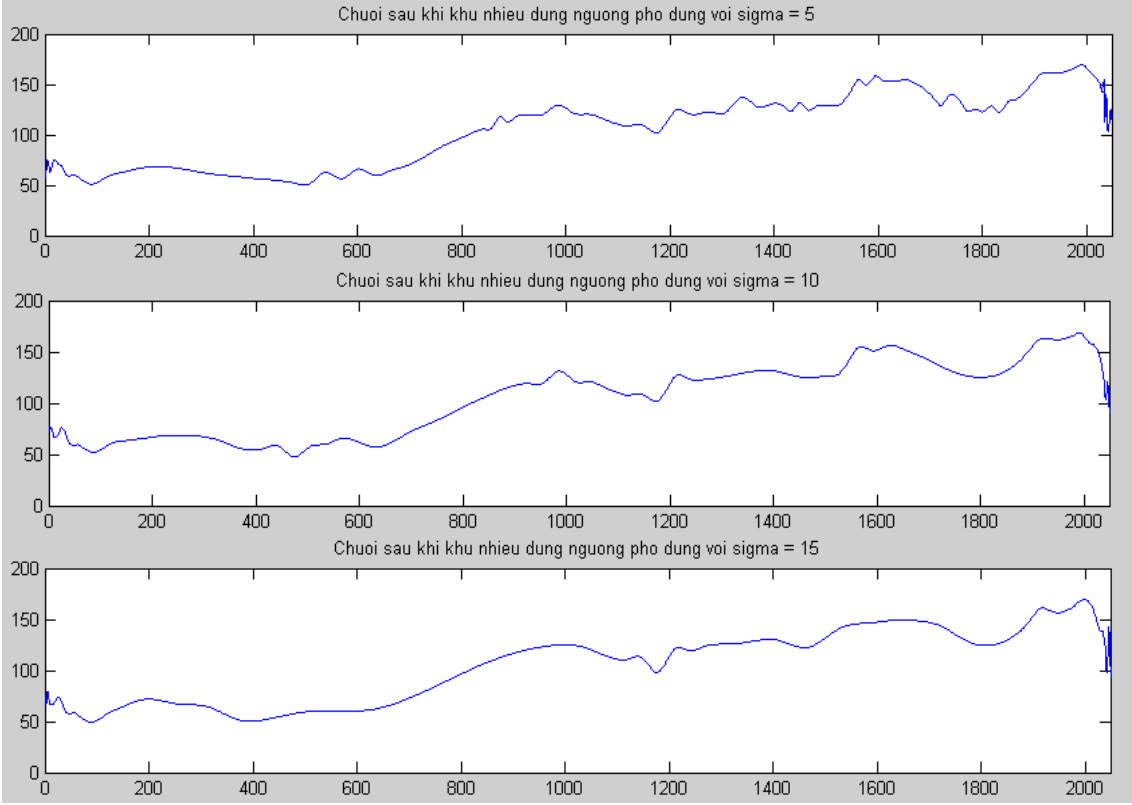
Hình 1. Chuỗi thời gian gốc và các chuỗi bị gây nhiễu với phương sai lần lượt là 5, 10, 15

Sau khi khử nhiễu bằng ngưỡng SURE đồ thị của các chuỗi đã được khử nhiễu cho ở hình 2 sau đây.



Hình 2. Các chuỗi thời gian sau khi khử nhiễu dùng ngưỡng SURE

Các chuỗi bị nhiễu sau khi đã được khử nhiễu bằng ngưỡng Phổ dụng cho ở hình 3 dưới đây.



Hình 3. Các chuỗi thời gian sau khi khử nhiễu dùng ngưỡng phổ dụng

Bảng sau đây cho biết các giá trị sai số của chuỗi bị nhiễu đối với chuỗi gốc, chuỗi sau khi khử nhiễu với chuỗi gốc với các phương sai nhiễu khác nhau.

Bảng 1. Bảng sai số

Loại ngưỡng	Sai số (%) \ Phương sai	$\sigma^2 = 5$	$\sigma^2 = 10$	$\sigma^2 = 15$
Ngưỡng SURE	$ME(X^{noise}, X^o)$	4.267	8.723	13.528
	$ME(X^{den}, X^o)$	2.219	4.819	9.248
	MRE	49.856	46.755	33.687
	$MSE(X^{noise}, X^o)$	0.125	0.265	0.399
	$MSE(X^{den}, X^o)$	0.067	0.160	0.299
Ngưỡng	$ME(X^{noise}, X^o)$	4.267	8.723	13.528

phổ dụng	$ME(X^{den}, X^{o})$	2.271	3.193	3.377
	MRE	46.325	61.339	76.053
	$MSE(X^{noise}, X^{o})$	0.125	0.265	0.399
	$MSE(X^{den}, X^{o})$	0.065	0.091	0.109

6. KẾT LUẬN

Qua bảng trên ta thấy ngưỡng SURE và phổ dụng chỉ khử nhiễu được khoảng 50% nhiễu về mặt trung bình, và khi phương sai nhiễu càng lớn thì ngưỡng phổ dụng khử nhiễu tốt hơn ngưỡng SURE.

Ta thấy rằng, các sai số là để đánh giá mức độ hiệu quả của từng biện pháp, phụ thuộc vào cách chọn hàm wavelet, phương sai của nhiễu, cách xác định ngưỡng hay đặc điểm của chuỗi thời gian.

Từ kết quả này chúng tôi thấy rằng công việc cần thiết tiếp theo là lập một độ đo sai số chuẩn cho khử nhiễu, giả sử là $\Delta\delta$, để sau khi khử nhiễu ta chỉ cần so sánh sai số tìm được với $\Delta\delta$ là có thể xác định được khử nhiễu có hiệu quả hay không. Ngoài ra tìm thêm các hàm ngưỡng để tăng tính hiệu quả của khử nhiễu.

APPLYING WAVELET METHOD FOR DENOISING IN TIME SERIES

To Anh Dzung, Hoang Van Ha
University of Sciences, VNU – HCM

ABSTRACT: *This paper presents an application of the Wavelet Method to denoise in time series, gives a compare with a traditional Fourier method, using the discrete transform. The key point of this method is choosing a wavelet function to estimate the variance of noise and determining the threshold. Moreover, we study some measures of the error and effectiveness among noise-suppression methods.*

TÀI LIỆU THAM KHẢO

[1]. Brani Vidakovic. *Statistical Modeling by Wavelet*. Jonh Wiley & Inc, (1999).

[2]. Donald B.Percival, Andrew T.Walden. *Wavelet Methods for Time Series Analysis*. Cambridge University Press, (2000).

[3]. C.Blatter. *Wavelet – A Primer*. AK Peters Natick, Massachusetts, (1998).

[4]. Bartosz Kozlowski. Time series denoising with wavelet transform. *Journal of Telecommunications and Information Technology*, 91 – 95, (2005).

[5]. Adhemar bultheel. *Wavelet with applications in signal and image processing*. CRC Press, (2003).

[6]. Carl Taswell. The What, How, and Why of Wavelet Shrinkage Denoising. *Computing in Science & Engineering*, (2000).

[7]. Time Series Data Library. Website: www-personal.buseco-monash.edu.au.