

Tipo de artículo: Artículo original  
Temática: Técnicas de programación  
Recibido: 25/01/2014 | Aceptado: 21/03/2014

## Esquema basado en Wavelet para la reducción de ruido online en señales industriales

### *Online wavelet-based scheme for reduction of noise in industrial signals*

Ing. Antonio Cedeño Pozo <sup>1\*</sup>, Dr. Rafael Trujillo Codorníu <sup>2</sup>

<sup>1</sup> Facultad 5. Universidad de las Ciencias Informáticas, Carretera a San Antonio de los Baños, km 2 ½, Torrens, Boyeros, La Habana, Cuba. CP.: 19370

<sup>2</sup> Universidad de Oriente, Ave. Patricio Lumumba s/n Altos de Quintero, Santiago de Cuba, CUBA. CP.:90500

\* Autor para correspondencia: [acedeno@uci.cu](mailto:acedeno@uci.cu)

---

#### Resumen

En este trabajo se propone un esquema para la reducción de ruido gaussiano online basado en wavelet con ventanas deslizantes para ser utilizado en señales industriales. Este tipo de técnicas se utilizan cuando la señal debe ser tratada a medida que se obtienen las muestras, funcionalidad necesaria en aplicaciones con requerimientos de tiempo real, por ejemplo en sistemas SCADA. El algoritmo introducido en el presente trabajo se diferencia de algoritmos similares en que: en primer término utiliza una selección adaptativa del umbral basada en NeighShrink; adicionalmente reutiliza en lo posible los coeficientes wavelet calculados en pasos anteriores para el procesamiento de cada punto; finalmente utiliza la wavelet Haar en lugar de otros filtros de mayor complejidad computacional. Para la comprobación experimental del nuevo esquema se utilizó un conjunto de señales sintéticas introducido por Donoho y Marron, complementadas por señales adquiridas en procesos reales de plantas industriales. Las pruebas realizadas revelan que el algoritmo propuesto mejora la reducción de ruido online respecto a esquemas similares reportados anteriormente y es computacionalmente menos complejo.

**Palabras clave:** reducción de ruido, señales industriales, transformada wavelet, ventanas deslizantes.

#### Abstract

*This paper proposes a scheme for online Gaussian noise reduction based on wavelets with sliding windows. The scheme has been specifically designed for real-time systems (e.g. SCADA systems) where signals must be processed on-the-fly. The proposed algorithm differs from previous proposals on: i) the adaptive threshold selection, which is based on NeighShrink; ii) the dynamic use of previously calculated wavelet coefficients and; iii) the use of the Haar wavelet filtering implementation instead of others with higher computational cost. The novel scheme has been tested on a set of pattern signals proposed by Donoho and Marron complemented with other representative measurements obtained from real processes. Our results indicate that, for this kind of data, the proposed algorithm outperforms previous schemes on both: noise reduction and efficiency.*

**Keywords:** industrial signals, moving windows, noise reduction, wavelet transform.

---

## Introducción

En las últimas dos décadas la Transformada Discreta de Wavelet (WDT) ha sido ampliamente utilizada para la reducción de ruido en diferentes tipos de señales (ver, (Yinping and Yongxing, 2012), (Jaffery *et al.*, 2010), (Haibing *et al.*, 2010)). La reducción de ruido offline basada en wavelet involucra tres pasos básicos: descomposición de la señal en sub-bandas wavelet mediante la WDT, modificación de los coeficientes wavelet (generalmente mediante la utilización de umbrales) y reconstrucción de la señal a partir de los coeficientes modificados.

De manera más precisa si se asume que los datos observados  $x(t) = s(t) + \eta(t)$  contienen la señal pura  $s(t)$  con un ruido aditivo blanco gaussiano  $\eta(t)$  como funciones muestreadas en el tiempo y mediante  $W(\cdot)$  y  $W^{-1}(\cdot)$  se denotan la transformada Wavelet directa e inversa respectivamente, entonces la técnica de reducción de ruido a la que hace referencia involucra los siguientes pasos:

1. Obtención de los coeficientes wavelet de la señal ruidosa

$$y = W(x) = \zeta + v; \quad \zeta = W(s); \quad v = W(\eta)$$

2. Modificación de los coeficientes obtenidos a partir de la aplicación de umbrales

$$Z = D(y, \lambda)$$

donde  $D(\cdot, \lambda)$  es un operador que modifica el valor del coeficiente wavelet a partir del umbral  $\lambda$ . Las umbralizaciones más conocidas fueron introducidas por Donoho y los operadores correspondientes son los siguientes:

$$D_s(u, \lambda) = \begin{cases} 0 & \text{si } |u| \leq \lambda \\ u - \lambda & \text{si } u > \lambda \\ u + \lambda & \text{si } u < -\lambda \end{cases}$$

$$D_h(u, \lambda) = \begin{cases} 0 & \text{si } |u| \leq \lambda \\ u & \text{si } u > \lambda \end{cases}$$

$D_s$  corresponde a la umbralización suave o flexible y  $D_h$  a la umbralización fuerte o fija. En este sentido se han publicado varias investigaciones que proponen modificaciones a estos tipos de umbralización (ver, por ejemplo, (Hongqiao and Shengqian, 2009), (Lin and Cai, 2010), (Liyuan *et al.*, 2010), (Chang *et al.*, 2010)).

### 3. Reconstrucción de la señal $s(t)$ mediante la transformada inversa

$$\mathfrak{s} = W^{-1}(Z)$$

El objetivo de los métodos de reducción de ruido es minimizar la relación  $s(t) - \mathfrak{s}(t)$  en la reconstrucción, o sea suprimir el ruido manteniendo, en lo posible, las propiedades de la señal original.

Donoho, considerado pionero en la reducción de ruido mediante la aplicación de umbrales, en (Donoho and Johnstone, 1995) introduce los métodos RiskShrink y VisuShrink, mientras que en (Donoho *et al.*, 1993) propone el esquema SureShrink. En particular, en el método VisuShrink, el valor del umbral  $\lambda$  es proporcional a la desviación estándar del ruido y depende de la longitud de la señal, tal y como se indica a continuación:

$$\lambda = \sigma_\eta \sqrt{2 \log(n)}$$

donde  $\sigma_\eta$  es la desviación estándar del ruido y  $n$  la longitud de la señal. La desviación estándar del ruido  $\sigma_\eta$  se puede estimar a partir de la señal ruidosa de diferentes maneras, uno de los estimadores más utilizados viene dado por la mediana de los valores absolutos de la primera banda  $H_1$  de descomposición wavelet:

$$\sigma_\eta = \frac{\text{median}(|y_i|, y \in H_1)}{0.6745}$$

En (Leigh *et al.*, 2011) se analizan otros estimadores del nivel de ruido y se discute la efectividad de cada uno de ellos en el contexto de la reducción de ruido.

El VisuShrink es catalogado dentro de los esquemas de umbral global pues el valor de  $\lambda$  se aplica a todos los coeficientes. Luego han aparecido métodos basados en la selección adaptativa del umbral, en los que el valor del

umbral varía teniendo en consideración las características espaciales de la señal analizada; dentro de estos métodos están los basados en reglas de Bayes, denominados BayesShrink (Chang *et al.*, 2000), Modified BayesShrink (Elyasi and Zarmehi, 2009), NormalShrink (Kaur *et al.*, 2002) y NeighShrink (Cai and Silverman, 2001).

El esquema NeighShrink propone que la umbralización se realice de manera individual para cada coeficiente wavelet considerando la influencia de los coeficientes colindantes. En este esquema el umbral utilizado para cada coeficiente wavelet se determina de acuerdo a la suma de los cuadrados de los coeficientes colindantes. Suponiendo que  $\{y_j\}$  es el conjunto de coeficientes wavelet de la señal ruidosa, si:

$$S_j^2 = \{y_{j-1}^2 + y_j^2 + y_{j+1}^2\}$$

es menor o igual que  $\lambda^2$ , entonces se establece  $y_j$  en cero. En caso contrario  $y_j$  se modifica de acuerdo a:

$$y_j = y_j \left( 1 - \frac{\lambda^2}{S_j^2} \right)$$

Se ha demostrado estadísticamente que los algoritmos de reducción de ruido *offline* mencionados son muy eficaces, sin embargo, no pueden ser aplicados tal cual en aplicaciones que tengan requerimientos de tiempo real. Por esa razón se han propuesto diferentes esquemas que tratan de adaptar estos algoritmos a esas situaciones. Estos esquemas realizan la reducción de ruido sobre la señal a medida que se van adquiriendo las muestras. La decisión de cómo se debe modificar el valor de la muestra, para suprimir el ruido, se realiza sólo sobre la base de los valores captados anteriormente de manera que no hay retardos a la salida. Para reducir el ruido de la última muestra obtenida se toma un conjunto de datos formado por la propia muestra y  $n$  muestras anteriores, de manera que a ese conjunto se le pueda aplicar el algoritmo de reducción de ruido *offline*. A ese conjunto de datos se le llama ventana y como la misma se modifica a medida que arriban nuevas muestras estos algoritmos se conocen como algoritmos de ventanas deslizantes.

En (Xia *et al.*, 2007) se propone un esquema de reducción de ruido *online* basado en ventanas deslizantes utilizando como método de umbralización el VisuShrink. La ventana se desliza cada vez que arriba una nueva muestra lo que implica que la WDT debe recalcularse en cada paso, lo cual es computacionalmente costoso, sobre todo si se tiene en cuenta que en ese trabajo se utiliza la *wavelet Symlet4*.

Hsiao-Ping Huang y Kuo-Yuan Luo proponen un algoritmo de reducción de ruido *online* basado igualmente en ventanas deslizantes (Huang and Luo, 2007). El esquema acondiciona la señal inicialmente utilizando el filtro de la

mediana. Como el algoritmo, a diferencia del mencionado anteriormente, no propone prolongación artificial de la ventana, es sensible a los efectos de borde en la estimación del valor de la muestra, por lo cual realiza una corrección final del valor utilizando el filtro de Kalman. Este esquema resulta más robusto que el propuesto en (Xia *et al.*, 2007) aunque a costa de una mayor complejidad computacional, que lo hace difícilmente utilizable en aplicaciones que adquieran, en tiempo real, un gran volumen de datos.

En el presente trabajo se propone un nuevo algoritmo que utiliza la selección adaptativa del umbral basada en NeighShrink y maneja las ventanas deslizantes de manera que puedan ser reutilizados, en lo posible, los coeficientes wavelet calculados en pasos anteriores para el procesamiento de cada punto. Adicionalmente se utiliza la *wavelet* Haar en lugar de otros filtros de mayor complejidad computacional, todo lo cual resulta en un algoritmo que se ejecuta alrededor de un 35 por ciento más rápido y logra una reducción del ruido más eficaz que el propuesto en (Xia *et al.*, 2007).

## Materiales y métodos

### Descripción del algoritmo propuesto

Como se mencionó anteriormente para reducir el ruido de la última muestra obtenida, que llamaremos muestra actual, se toma un conjunto de datos formado por la propia muestra y  $n$  muestras anteriores, de manera que a ese conjunto se le pueda aplicar el algoritmo de reducción de ruido *offline*. Para evitar que los efectos negativos que ocurren al aplicar el esquema *offline* en los extremos de la ventana afecten a la reducción del ruido en la muestra actual, la ventana se prolonga artificialmente hacia el futuro, completando simétricamente (en forma de espejo) los valores desconocidos que aún no han arribado (Ver Figuras 1(a) y 1(b)).

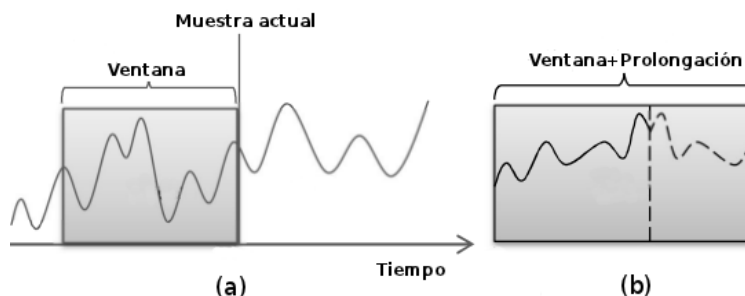


Figura 1. (a) Ventana deslizante. (b) Prolongación de la ventana.

La ventana alrededor de la muestra  $x(i)$  se define como un conjunto de datos de la siguiente forma:

$$W = \begin{cases} \{\text{Vacío}\} & \text{si } i < l \\ \{x(i-l+1), x(i-l+2), \dots, x(i)\} & \text{si } i \geq l \end{cases}$$

donde  $x(i)$  representa la muestra de índice  $i$  de la señal en tiempo real y  $l$  la longitud de la ventana de datos. Como se mencionó anteriormente para evitar efectos indeseados como la distorsión en los bordes y el fenómeno de Gibbs (Kelly, 1996) se utiliza una extensión de la ventana, que en este material se denomina prolongación. La cantidad de elementos de la prolongación  $p$  fija la longitud total de la ventana extendida en  $l + p$ . De esta forma la ventana extendida se expresa mediante:

$$W = \begin{cases} \{\text{Vacío}\} & \text{si } i < l \\ \{x(i-l+1), x(i-l+2), \dots, x(i), x(i), \dots, x(i-p)\} & \text{si } i \geq l \end{cases}$$

Uno de los aspectos que hacen de esquema de ventanas deslizantes, que en lo adelante se denominará MW, un mecanismo complejo computacionalmente es la necesidad de recalcular los coeficientes *wavelets* ante la llegada de cada muestra. En la presente investigación los autores proponen una modificación a la definición de la ventana tradicional de MW, con el objetivo de poder reutilizar los coeficientes *wavelets* calculados al arribo del punto anterior tanto como sea posible.

En MW se utiliza una prolongación de tamaño fijo y al llegar una muestra nueva la ventana se desliza en un elemento. En el que esquema que se propone, que llamaremos de ventanas deslizantes por bloques (MWB), el tamaño de la prolongación es variable y siempre que sea posible, el inicio de la ventana, no se desliza ante la llegada de una nueva muestra. Inicialmente el tamaño de la prolongación es igual al parámetro  $p_{max}$ . A medida que arriban las muestras, para mantener constante el tamaño de la ventana extendida, se requiere que la longitud de la prolongación  $p$  vaya decreciendo de uno en uno hasta que llegue a un valor mínimo prefijado  $p_{min}$ . Consecuentemente la longitud de la ventana  $l$  debe incrementarse de uno en uno. Al conservarse el inicio de la ventana todos los coeficientes wavelet, calculados previamente y que se encuentran alejados de la muestra actual, pueden reutilizarse.

Se puede deducir, que para la transformada Haar, la cantidad  $m_j$  de coeficientes wavelets en el nivel de descomposición  $j$  que pueden ser reutilizados se obtiene a partir de la siguiente expresión:

$$m_j = \begin{cases} \{(m_{j-1}) / 2\} & \text{si } m_{j-1} \text{ es par} \\ \{(m_{j-1} - 1) / 2\} & \text{si } m_{j-1} \text{ es impar} \end{cases}$$

donde  $m_0 = l - 1$ . Cuando el tamaño de la prolongación es el mínimo configurado y arriba una nueva muestra la ventana extendida se desliza en un bloque de datos de longitud  $p_{max} - p_{min}$ . En resumen el algoritmo que se propone puede sintetizarse en los siguientes pasos:

1. Inicialización.

- Se definen los siguientes parámetros: tamaño mínimo de la ventana  $l_{min}$  y tamaños mínimos y máximos de la prolongación ( $p_{min}$  y  $p_{max}$ ). Se sugiere que la suma  $l_{min} + p_{max}$  sea un múltiplo de  $2^k$  donde  $k$  es el número máximo de niveles de descomposición deseados.
- Se inicializa  $l = l_{min}$ ,  $p = p_{max}$  y el número de muestras ya almacenadas en la ventana  $s = 0$ .

2. Fase de espera.

- Mientras  $s < l$  las muestras se devuelven tal cual y se incrementa  $s$  al arribo de cada una de ellas. Si  $s \geq l$  se pasa directamente a la fase siguiente.

3. Fase de reducción de ruido.

Al arribo de cada muestra  $x_i$ :

- Se hace  $l = l + 1$ ,  $p = p - 1$ .
- Si  $p < p_{min}$  entonces se desliza el inicio de la ventana en  $p_{max} - p_{min}$  elementos y se restablece  $p = p_{max}$  y  $l = l_{min}$ . En caso contrario se continúa en el próximo paso.
- Se prolonga la ventana de datos en forma de espejo tal y como se indica en la Figura 1.
- Se aplica la transformada Haar calculando aquellos coeficientes que no se reutilizan. Si la ventana se desplazó en  $p_{max} - p_{min}$  elementos se calculan todos los coeficientes wavelet, en caso contrario se copian los almacenados para el punto anterior de acuerdo a la fórmula de  $m_j$  y se recalcula el resto.
- Se realiza la umbralización sobre los coeficientes *wavelets* resultantes utilizando el algoritmo NeighShrink.
- Se restauran los datos de la ventana mediante la transformada Haar inversa.
- Se devuelve el valor  $x_i$  resultante.

## Diseño experimental

En los experimentos se utilizaron dos grupos de señales, el primero lo componen 11 señales sintéticas propuestas por Donoho (Donoho *et al.*, 1993) y Marron (Marron *et al.*, 1995), algunas de las cuales se muestran en la Figura 2, mientras que el segundo grupo está formado por 16 señales tomadas de procesos reales monitoreados por un sistema SCADA, en la Figura 3 se puede observar una representación de este segundo grupo. En todos los casos se utilizan

2048 muestras y algunas de esas señales fueron escaladas para que los intervalos de variación fueran similares. La naturaleza de las señales industriales es muy amplia y contiene mediciones de temperaturas, presiones, niveles, corriente, flujos, etc. En la selección que se hace se distinguen grupos de señales con diferentes características, señales casi constantes, lineales a tramos, suaves a tramos e impulsivas.

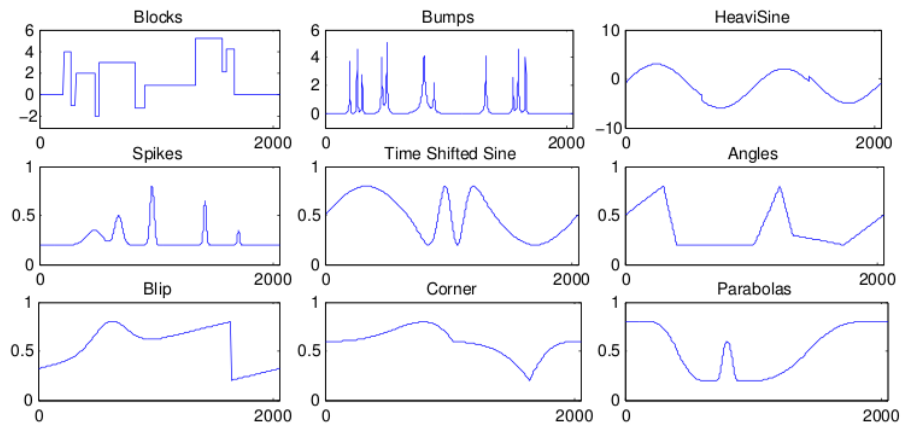


Figura 2. Señales sintéticas.

Dada la enorme cantidad de procesos que pueden encontrarse dentro de las industria, es muy difícil seleccionar un conjunto de señales que cubran en su totalidad las características que pueden presentar las señales industriales, sin embargo, los autores de esta investigación no han querido circunscribirse solamente a las señales sintéticas comúnmente utilizadas en otros trabajos. Estas señales sintéticas garantizan la reproducibilidad de la experiencia, pero la inclusión de señales reales con diversas características brinda una visión más realista y complementa a las señales sintéticas conformando un conjunto que se considera adecuado para las evaluaciones de los algoritmos en cuestión.



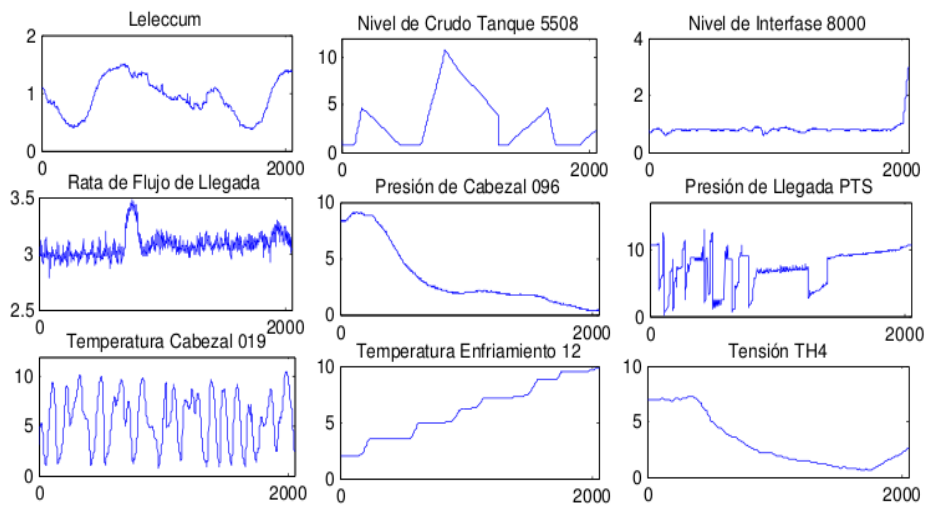


Figura 3. Señales reales.

En las pruebas realizadas se adiciona ruido gaussiano con varianzas  $\sigma_2 = 0.01$ ,  $\sigma_2 = 0.03$ ,  $\sigma_2 = 0.05$ ,  $\sigma_2 = 0.07$  y  $\sigma_2 = 0.09$ . La métrica que se utilizó para la evaluación de los algoritmos fue el Error Medio Cuadrático (MSE), representado por:

$$MSE = \frac{1}{N} \sum_{i=1}^N (X - \hat{X})^2$$

donde  $N$  es la cantidad de muestras de la señal, el vector  $X$  representa la señal original y  $\hat{X}$  representa a la señal una vez tratada con los algoritmos propuestos.

Se realizaron 200 corridas de ambos algoritmos para cada nivel de ruido, lo que arroja 270 grupos de muestras de MSE de 200 valores cada una, a partir de  $2 \text{ Algoritmos} * 27 \text{ Señales} * 5 \text{ Niveles de ruido}$ . En el diseño experimental se incluye la medición del tiempo de ejecución total, pues en el caso de los algoritmos *online* este indicador resulta de mucho interés.

Los parámetros utilizados en el experimento fueron los siguientes:

- $l_{min} = 192$
- $p_{min} = 6$
- $p_{max} = 64$

Con estos parámetros se garantiza que el tamaño de la ventana extendida para los dos algoritmos sea de 256 elementos.

Para verificar si la diferencia de eficacia entre los algoritmos que intervienen en el experimento es estadísticamente significativa se utiliza la prueba de hipótesis paramétrica de Student (ver, por ejemplo, Zimmerman (1997)). Esta prueba aplica al caso en el que los grupos muestrales pertenezcan a una distribución normal y las comparaciones se realicen solamente entre dos algoritmos. Mediante la aplicación de la prueba de Kolmogorov-Smirnov (ver, por ejemplo, Marsaglia *et al.* (2003)) se comprobó el supuesto de la normalidad en las muestras de MSE, lo que unido a la independencia de las mediciones garantiza el empleo satisfactorio de la prueba de hipótesis Student.

La hipótesis nula para este experimento queda definida como:

$$H_0: \mu_1 = \mu_2$$

donde  $\mu_1$  y  $\mu_2$  representan las medias de grupos muestrales de los algoritmos que se analizan, al ser aplicados sobre una señal del conjunto patrón, y con el mismo nivel de ruido. La hipótesis alternativa se define como:

$$H_1: \mu_1 \neq \mu_2$$

Igual tratamiento se realizó para verificar si las diferencias en el tiempo de procesamiento de los algoritmos mencionados es estadísticamente significativa.

## Resultados y discusión

En las Figuras 4, 5 y 6 se representan gráficamente las curvas de MSE promedio contra diferentes valores de varianza del ruido. Visualmente se evidencia la diferencia entre los algoritmos en cuestión para todas las señales. El algoritmo propuesto, representado por la curva de color rojo, obtiene valores inferiores de MSE promedio en todas las señales. De forma empírica se puede afirmar que el algoritmo propuesto tiene mejor rendimiento sobre el conjunto de señales industriales definido en la investigación como representativo de las señales que se generan en procesos industriales. Para demostrar que la diferencia de rendimiento es estadísticamente significativa se aplica la prueba de Student sobre la base de las hipótesis definidas anteriormente con un 95 por ciento de confianza. En todos los casos las diferencias observadas resultaron significativas.

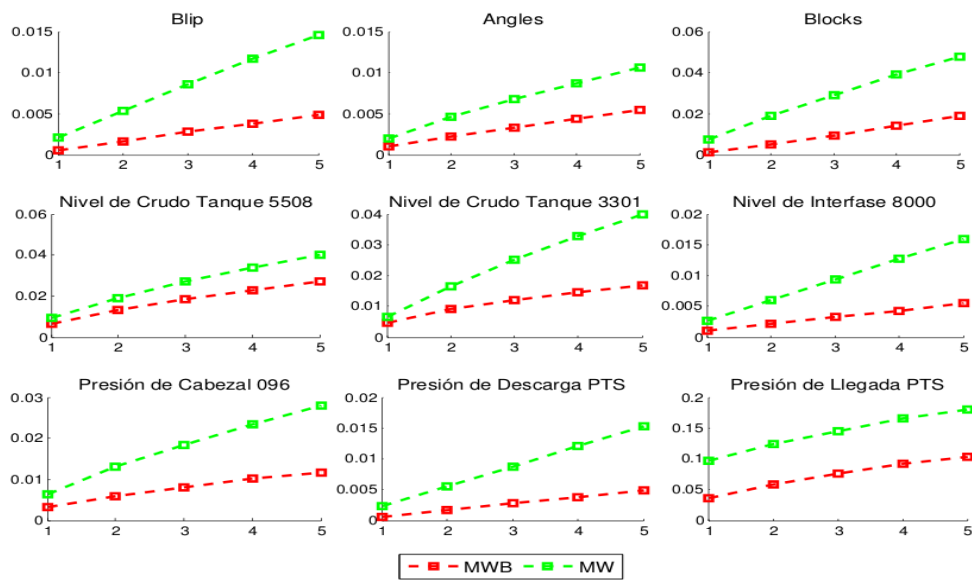


Figura 4. Gráficas de MSE promedio para cinco niveles de ruido ( $\sigma^2=0.01$ ,  $\sigma^2=0.03$ ,  $\sigma^2=0.05$ ,  $\sigma^2=0.07$ ,  $\sigma^2=0.09$ ).

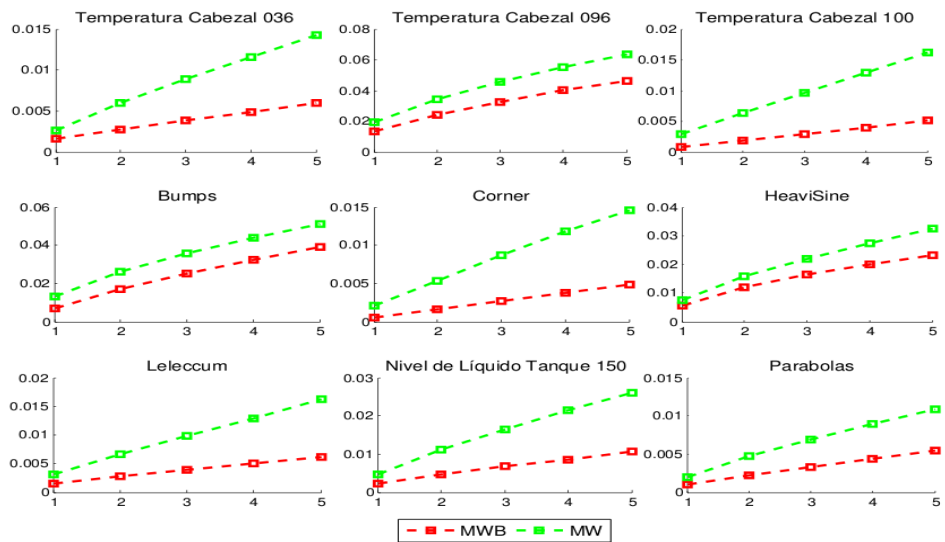


Figura 5. Gráficas de MSE promedio para cinco niveles de ruido ( $\sigma^2=0.01$ ,  $\sigma^2=0.03$ ,  $\sigma^2=0.05$ ,  $\sigma^2=0.07$ ,  $\sigma^2=0.09$ ).

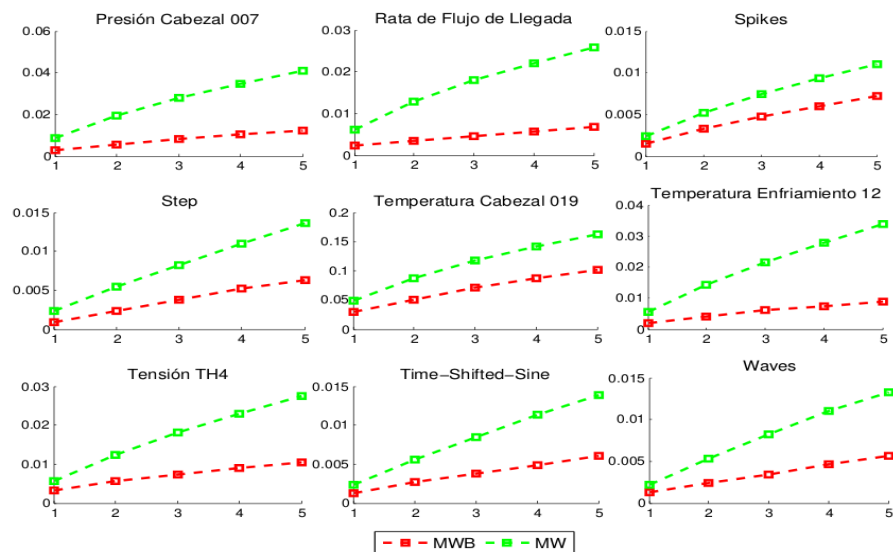


Figura 6. Gráficas de MSE promedio para cinco niveles de ruido ( $\sigma^2=0.01$ ,  $\sigma^2=0.03$ ,  $\sigma^2=0.05$ ,  $\sigma^2=0.07$ ,  $\sigma^2=0.09$ ).

Como se ha mencionado el tiempo de ejecución es una cuestión muy importante en el procesamiento online, en la Figura 7 se puede constatar la notable diferencia entre los tiempos de ejecución de ambos algoritmos. Las barras rojas representan los tiempos para el algoritmo propuesto en el procesamiento de las señales del conjunto patrón, y las azules los tiempos del esquema propuesto en (Xia *et al.*, 2007). Como promedio el nuevo algoritmo se ejecuta alrededor de un 35 por ciento más rápido.

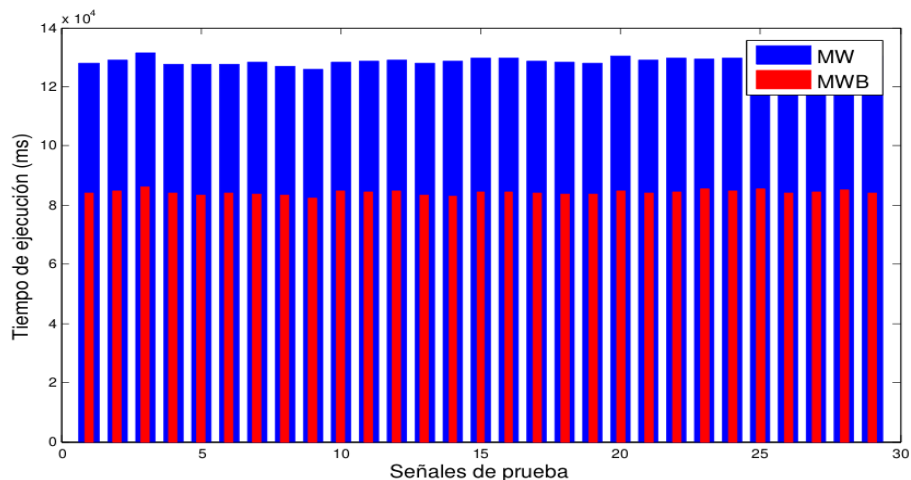


Figura 7. Tiempos de ejecución en milisegundos para los algoritmos analizados.

## Conclusiones

Los resultados obtenidos, a partir de las pruebas realizadas que se muestran en las figuras 4, 5, 6 y 7, indican que el esquema propuesto para la reducción de ruido *online* en señales industriales, se ejecuta alrededor de un 35 por ciento más rápido y logra una reducción del ruido más eficaz que el propuesto en (Xia *et al.*, 2007), tanto sobre el conjunto de señales sintéticas definidas por Donoho y Marron, como en las señales obtenidas de procesos reales.

## Referencias

- CAI, T. T., SILVERMAN, W., 2001. Incorporating information on neighbouring coefficients. *Sankhya: The Indian Journal of Statistics* 63, 127–148.
- CHANG, FENGXIANG, HONG, WENXUE, ZHANG, TAO, JING, JUN, LIU, XULONG, 2010. Research on wavelet denoising for pulse signal based on improved wavelet thresholding. *Pervasive Computing, Signal Processing and Applications, International Conference 0*, 564–567.
- CHANG, S. G., YU, B., VETTERLI, M., 2000. Adaptive wavelet thresholding for image denoising and compression. *IEEE Transactions on Image Processing* 9 (9), 1532–1546.
- DONOHO, D., JOHNSTONE, I., JOHNSTONE, I. M., 1993. Ideal spatial adaptation by wavelet shrinkage. *Biometrika* 81, 425–455.
- DONOHO, D., JOHNSTONE, I. M., 1995. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association* 90, 1200–1224.
- ELYASI, I., ZARMEHI, S., 2009. Elimination noise by adaptive wavelet threshold. *World Academy of Science, Engineering and Technology* 56, 56–86.
- JAFFERY, Z. A., AHMAD, K., AFROZ, 2010. Performance comparison of wavelet threshold estimators for ecg signal denoising. *Advances in Recent Technologies in Communication and Computing, International Conference 0*, 248–251.
- HAIBING, QI, XIONGFEI, LIU, CHAO, PAN, 2010. Discrete wavelet soft threshold denoise processing for ecg signal. *Intelligent Computation Technology and Automation, International Conference 2*, 126–129.
- HONGQIAO, Li, SHENGQIAN, WANG 2009. A new image denoising method using wavelet transform. *Information Technology and Applications, International Forum 1*, 111–114.
- HUANG, H.-P., LUO, K.-Y., 2007. On-line wavelets filtering with application to linear dynamic data reconciliation. *Industrial and Engineering Chemistry Research* 46 (25), 8746–8755.

- KAUR, L., GUPTA, S., CHAUHAN, R. C., 2002. Image denoising using wavelet thresholding. In: Indian Conference on Computer Vision, Graphics and Image Processing, Ahmedabad.
- KELLY, S. E., 1996. Gibbs phenomenon for wavelets. *Applied and Computational Harmonic Analysis* 3 (1), 72–81.
- LEIGH, A., WONG, A., CLAUSI, D. A., FIEGUTH, P., 2011. Comprehensive analysis on the effects of noise estimation strategies on image noise artifact suppression performance. *Proceedings of the IEEE International Symposium on Multimedia ISM '11*, 97–104.
- LIN, Y., CAI, J., 2010. A new threshold function for signal denoising based on wavelet transform. *Measuring Technology and Mechatronics Automation, International Conference* 1, 200–203.
- LIYUAN, M., YONGGANG, D., YONGJUN, L., TIANHUI, W., 2010. Improved algorithm for denoising based on wavelet threshold and performance analysis. *Pervasive Computing, Signal Processing and Applications, International Conference* 0, 572–575.
- MARRON, J. S., ADAK, S., JOHNSTONE, I. M., NEUMANN, M. H., PATIL, P., 1995. Exact risk analysis of wavelet regression.
- MARSAGLIA, G., TSANG, W., WANG, J., 2003. Evaluating kolmogorov's distribution. *Journal of Statistical Software* 8 (18).
- XIA, R., MENG, K., QIAN, F., ZHEN-LEI, W., 2007. Online wavelet denoising via a moving window. *Acta Automatica Sinica* 33 (9), 897–901.
- YINPING, MA, YONGXING, HUANG, 2012. Adaptive threshold based on wavelet transform fingerprint image denoising. *Computer Science and Electronics Engineering, International Conference* 3, 494–497.
- ZIMMERMAN, D. W., 1997. A note on interpretation of the paired-samples t test. *Journal of Educational and Behavioral Statistics* 22 (3), 349-360.