USE OF WAVELET TRANSFORM FOR SIGNAL DENOISING OF NUCLEAR POWER PLANT

J.-M. LINA[1,2], H. DAI[2], W. THOMPSON[2] AND B. GOULARD[1]

1: Centre de Recherches Mathematiques, Univ. de Montréal
C.P. 6128 Succ. Centre-Ville, Montréal (Quebec), H3C 3J7, Canada

2: Atlantic Nuclear Services Ltd., Fredericton, New Brunswick, E3B 5C8, Canada

Abstract

A tool that reliably removed or reduced the noise from signals in nuclear processes would help considerably in monitoring the condition of safety related or process signals. In this paper, we describe wavelet-based filtering technologies, and how they have been developed and integrated into an existing Windows software package, the Plant Analysis Workbench (PAW). The added feature using the wavelet technology in PAW has been used to denoise real-time data collected in PLGS. Compared with the traditional filtering technology, the wavelet filtering technology in PAW can produce much more accurate and satisfactory results. This paper gives a complete description of the use of the wavelet shrinkage technique underlying the denoising algorithm used in PAW.

1. INTRODUCTION

One of the basic problems in the validation of data from nuclear power plant is how to estimate signals from noise-contaminated data. Traditional filtering techniques based on Fourier representation of the data are not appropriate for characterizing strong transients in signals since each mode of the Fourier decomposition contains information that describes a particular localized feature in the signal. In contrast with the trigonometric functions, wavelets can be supported on a finite interval. Therefore, combining the traditional filtering technologies with the wavelet transform creates a powerful near optimal approach for denoising contaminated signals which may have complex transient responses.

In this paper, we describe a filtering technique based on a complex valued discrete wavelet transform. This technology has been built into a Windows-based software package, Plant Analysis Workbench (PAW), used routinely for operations at Point Lepreau Generating Station (PLGS). It has been shown that this filtering technology is very powerful in removing additive noise from contaminated signals' spectrum changes within a specific time period. In fact, the developed wavelet-based filtering technology has been used successfully during the recent start-up operations at PLGS.

The paper is organized as follows. In the next section, we present the basics of the wavelet-based multiresolution analysis of a signal. Section 3 describes the filtering technique with some mathematical arguments for justifying it. Integration of the wavelet technology for signal denoising into PAW and applications of the wavelet denoising feature in PLGS are given in Section 4. The last section, Section 5, gives a summary of the paper.

2. WAVELETS AND MULTIRESOLUTION ANALYSIS

A multiresolution analysis of a signal $x(t)$ is a sequence of approximation spaces $V_j \subset L^2(\mathbb{R})$,

$$\ldots \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \ldots \tag{1}$$

and $P_j x(t)$ represents the projection of the signal onto V_j so that this projection is the closest approximation of $x(t)$ with resolution 2^{-j}. Each space V_j is generated through the discrete translation of a scaling function φ scaled at the appropriate resolution, that is

$$P_j x(t) = \sum_k c_{j,k} \varphi_{j,k}(t) \tag{2}$$

where

$$\varphi_{j,k}(t) = 2^{j/2} \varphi(2^j t - k) \tag{3}$$

The coefficients $c_{j,k}$ are labelled with a position index k and scale index j: the larger is the value of j, the more squeezed is the function $\varphi_{j,k}(t)$. The detail signal at the resolution 2^{-j} is defined by the difference of two subsequent approximations,

$$Q_j x(t) = P_{j+1} x(t) - P_j x(t) \tag{4}$$

and furthermore, the signal can be expanded as

$$Q_j x(t) = \sum_k d_{j,k} \psi_{j,k}(t) \tag{5}$$

where

$$\psi_{j,k}(t) = 2^{j/2} \psi(2^j t - k) \tag{6}$$

is the wavelet function scaled at the resolution of the details and localized at position k. In her seminal work, Daubechies defined such basic functions subjected to the following constraints: optimal localization in both space and frequency and orthogonality of the whole set of the discrete dilations and translations of the genuine function ψ. In the present work, we use an extension of her construction by considering "symmetric compactly supported orthogonal wavelets". The major difference with the standard Daubechies wavelets is the complex value of ψ (and of the associated scaling function ψ) implied by the symmetry property. Figures 1 and 2 display such complex functions. The symmetry property is worth introducing because it greatly reduces the shift variance involved in general with the discrete wavelet representation of signals.

Given a signal $x(t)$ regularly sampled at time t_i, $i = 1, \ldots, 2^m$, we first consider the finer approximation of the signal $P_m x(t)$. The wavelet representation of the signal is the decomposition

$$P_m x(t) = P_{m_o} x(t) + \sum_{j=m_o}^{m-1} Q_j x(t) \tag{7}$$

and the discrete multiresolution analysis of $x(t)$ consists of the computation of the coefficients of the expansion

$$P_m x(t) = \sum_k c_{m_o k} \varphi_{j,k}(t) + \sum_{j=m_o}^{m-1} \sum_k d_{j,k} \psi_{j,k}(x) \tag{8}$$

In this expansion, $j = m_o < m$ is an arbitrary low resolution scale. The coefficients in the previous expansion are computed through the orthogonal projection of the field upon the multiresolution basis:

$$\begin{cases} c_{j,k} = \int \overline{\varphi_{j,k}(t)} f(t)\, dt \\ d_{j,k} = \int \overline{\psi_{j,k}(t)} f(t)\, dt \end{cases} \tag{9}$$

Writing the basic decomposition $P_{j-1}x(t) = P_jx(t) + Q_jx(t)$ in terms of the wavelets modes, the fast wavelet decomposition transform (FWT) is the algorithm composed with the low-pass filter (α_k) and the high-pass filter ($b_k = (-1)^k\ \overline{a_{1-k}}$) associated with the two projectors P_j and Q_j, respectively:

$$\begin{cases} c_{j,n} = \sqrt{2} \sum_k \bar{a}_{k-2n}\, c_{j+1,k} \\ d_{j,n} = \sqrt{2} \sum_k \bar{b}_{k-2n}\, c_{j+1,k} \end{cases} \tag{10}$$

Starting from the initial set of coefficients $c_{m,k}$ that represent the data at the finer resolution, the iterative action of the previous algorithm gives the coefficients of the coarser resolution ($c_{m_0,k}$) at subsequent dyadic scales, $j = m_0, m_0 + 1, \ldots, m - 1$.

Conversely, the reconstruction algorithm is expressed by the inverse FWT:

$$c_{j+1,n} = \sqrt{2} \sum_k a_{n-2k}\, c_{j,k} + \sqrt{2} \sum_k b_{n-2k}\, d_{j,k} \tag{11}$$

In the present work, the filters are complex valued and symmetric: $a_k = a_{1-k}$. The next table displays the filter coefficients associated with the complex wavelet of Fig. 1.

k	a_k	φ_{k-1}
1	$0.662912 + 0.171163i$	$0.976562 - 0.413521i$
2	$0.110485 - 0.085581i$	$0.015624 + 0.221889i$
3	$-0.066291 - 0.085581i$	$-0.003906 - 0.015128i$

Let us conclude this Section with a remark about the initialization of the FWT. Starting with samples of a signal, say x_i, we need to estimate carefully the coefficients of the finest approximation $P_mx(t)$. Here again, the symmetry of the basis is helpful and we can easily show that, defining $\phi_k = \phi_{-k} = \phi((2k+1)/2)$, we have an accurate approximation given by

$$P_mx(t) = \sum_k c_{m,k}\varphi_{m,k}(t) \qquad \textit{with} \quad c_{m,k} = \sum_n \overline{\Phi_n} x_{k-n} \tag{12}$$

To summarize, the wavelet transform is the mapping

$$\{c_{m,k}\} \rightarrow \{c_{m_0,k}, d_{m_0,k}, d_{m_0+1,k}, \dots, d_{m-1,k}\} \tag{13}$$

All the quantities are complex valued. Conversely, the inverse wavelet transform is the mapping

$$\{c_{m_0,k}, d_{m_0,k}, d_{m_0+1,k}, \dots, d_{m-1,k}\} \rightarrow \{c_{m,k}\} \tag{14}$$

and the reconstruction of the signal is given by

$$x_i = \sum_k c_{m,k}\, \varphi_{m,k}(t_i) \quad i.e. \quad x_i = \sum_k c_{m,i-k}\, \phi_k \tag{15}$$

3. WAVELET SHRINKAGE AND NON-LINEAR APPROXIMATION

We suppose that we are interested in a function $x(t)$ for which we know a regular sampled time sequence corrupted with an additive Gaussian white noise:

$$y_i = x_i + \sigma \varepsilon_i, \qquad i = 1, 2, \ , 2^m \tag{16}$$

Donoho and Johnstone proposed a three steps method for recovery of $x(t)$:

(a) Perform a multiresolution analysis from the empirical data y_i yielding the noisy wavelet coefficients $d_{j,k}$, $j = m_0, m_0 + 1, \dots, m - 1$ and $k = 1, 2, \dots, 2^j$.

(b) Given a threshold λ (defined later), apply the non-linear soft-thresholding operator to the noisy wavelet coefficients:

$$d_{j,k} \rightarrow s_\lambda(d_{j,k})$$

with

$$s_\lambda(z) = (1 - \frac{\lambda}{|z|})z \quad if\ \lambda < |z|,\ and\ 0\ elsewhere \tag{18}$$

(c) Invert the wavelet transform with the threshold wavelet coefficients and estimate the signal from the new coefficients $c_{m,k}$.

$$\hat{x}(t_i) = \sum_k \overline{c_{m,i-k}}\, \phi_k \tag{19}$$

The choice of the threshold is critical: a large value of λ gives a wavelet estimator that underfits the data (large bias); small value of λ gives overfitting (large variance). Donoho and Johnstone proposed different policies for choosing λ. Here, we adopt the so-called universal visu shrinkage for which

$$\lambda = \hat{\sigma} \sqrt{2 \log 2^m}$$

where $\hat{\sigma}$ is an estimate of the noise level. An important feature of this choice is that it guarantees a noise-free reconstruction of the signal.

Let us conclude the Section with few remarks and a test experiment.

- How can we estimate $\hat{\sigma}$? The noise level can be accurately computed from the modulus of the wavelet coefficients at the finer scale. This is because only few wavelets coefficients $d_{m-1,k}$ of the first level of the wavelet transform are relevant for the true signal. Conversely, all the coefficients are corrupted by noise. This observation justifies a simple statistical measurements of the noise level directly from the amplitude of the $d_{m-1,k}$. This estimate is given by

$$\hat{\sigma} = \frac{median(|d_{m-1,k}|)}{0.645} \tag{20}$$

- Why does it work? One important feature of the wavelet bases is that they provide unconditional bases of a wide range of smoothness spaces. This means that the various smoothness measurements can be directly computed from the wavelet coefficients. The wavelet shrinkage acts as a smoothing operation in any of this range of smoothness measures. Let us note at this point that the shrinkage defined by (18) preserves the phase of the coefficients. It has been shown that the phases of the wavelet coefficients contains important information about the local transients in the signal. The amplitude of the coefficients can be interpreted as the probability of occurrence of such transients in the desired signal.

- Experiment with a real signal was conducted and results are provided in Figures 3 and 4. Figure 3 displays 1024 samples of data contaminated with noise. The solid line in the figure is the coarse approximation of the signal hidden in those data. The approximation is $P_m x(t)$ given in Equation (7) with $m\text{-}m_0=7$ (i.e., 7 levels in the wavelet decomposition) and $m=10$. Figure 4 shows the denoised signals after the action of the shrinkage technique. During the synthesis of this signal, 87% of the original wavelet coefficients have been shrunk to 0 because they were associated with noise.

4. WAVELET DENOISING FEATURE IN PAW

From the engineering point of view, wavelets can be considered as scales according to which the functions or sampled data are analyzed. Wavelets can be supported on almost any arbitrarily small interval. By processing the data at different scales, wavelets give a representation of the signal that extract local "details" of the signal. In practice, these "details" are computed from two parallel convolutions, a low-pass and a high-pass filtering respectively.

Given an empirical signal, its multi-resolution analysis amounts to projecting it in successive "detail" spaces associated with scales from the finer (initial sampling resolution) to some coarser scales. The noise component of a signal is projected in the "detail" spaces. The "shrinkage" technique, which is used to denoise the signal, consists in defining a set of thresholds and a threshold rule such that scale by scale, the noise can be removed by "killing or preserving" the wavelet coefficients.

4.1 Wavelet Denoising Feature

The wavelet filtering technology has been integrated into an earlier developed easy-to-use tool, the Plant Analysis Workbench (PAW) [6], for the purpose of removing noise components contained in sampled signals. The wavelet denoising feature of PAW has been tested and used by engineers in PLGS.

The first successful use of the wavelet denoising feature was during start-up operations in PLGS at the end of December 1996. One of the important start-up operations in PLGS was to detect possible channel flow blockage caused by small pieces of wood in the channels. The flow blockage can be detected by identifying the shape of a step response, the channel outlet temperature, to a step input, the channel inlet temperature, in the reactor. An extra delay in the step response represents a possible flow blockage. However, the identification of such a delayed step response cannot be accurately done by directly using the contaminated data. The identification package for flow blockage gives lots of false alarm of the flow blockage when data contain unnecessary variations due to the noise component. During the start-up operation, the Wavelet filter was used, as a means of signal preprocessing, to denoise the sampled data. Then, the denoised data was sent to the identification package to detect the possible flow blockage. It has been proved that the flow blockage can be successfully detected by using the signals denoised using the Wavelet filter.

In addition to the above usage, the denoising feature has been widely applied to other situations. The results will be shown in Section 4.2.

There are four ways of generating Wavelet Denoised signals using PAW. These procedures are summarized as follows:

Procedure 1. Apply the denoising feature to a set of redundant signals, or a selected signal by clicking the "***Display***" menu item:

- Step 1.1: From the PAW main menu, click "***Display***" item to show a DISPLAY dialogue box;
- Step 1.2: Select an interested VARIABLE after selecting a proper GROUP;
- Step 1.3: Click the "***Wavelet***" button within MODELS block of the dialogue box;
- Step 1.4: Enable the "***Over Write***" option as if necessary by clicking "***Overwrite***" option;
- Step 1.5: Click the "***Display***" button to view the denoised signals on the screen. If the denoised signals do not exist, PAW will automatically launch the wavelet denoising DLL for the given signals. Clicking the "***Cancel***" button will leave the dialogue box without performing the wavelet denoising operation;
- Step 1.6: Optionally, select one of printing sub-menus under the "***Print***" pull-down menu within the "***File***" main menu item to print the graph.

Procedure 2. Apply the denoising feature to a set of redundant signals, or a selected signal by clicking the "***Wavelet Denoise***" item under the "***Function***" menu item:

- Step 2.1: From the PAW main menu, click the "***Function***" item to show a pull-down menu, and then select the "***Wavelet Denoise***" item to activate a Wavelet Denoising dialogue box;
- Step 2.2: Select an interested VARIABLE after selecting a proper GROUP;
- Step 2.3: Click the "***OK***" button to perform wavelet denoising of the selected signals and view the denoised signals on the screen. Clicking the "***Cancel***" button will leave the dialogue box without performing the wavelet denoising operation;
- Step 2.4: Optionally, select one of printing sub-menus under the "***Print***" menu item within the "***File***" pull-down menu to print the graph.

Procedure 3. Apply the denoising feature to a set of selected variables which by clicking the "***Build***" menu item within the "***File***" pull-down menu:

- Step 3.1: From the PAW main menu, click the "***File***" item to show a pull-down menu, and then select the "***Build***" item to activate a BUILD dialogue box;
- Step 3.2: Select a norm file for "***Previous Norm File***". Usually, a previous norm file can be found in a directory like "*c:\paw\longterm*". The default file is satisfactory for use with Wavelets;
- Step 3.3: Select interested VARIABLEs from a list of GROUPs;
- Step 3.4: Click the "***Wavelet***" button within MODELS block of the dialogue box;

Step 3.5: Enable the "*Over Write*" and "*Global Removal*" options as if necessary by clicking the options. Both options are enabled as default;

Step 3.6: Click the "*Build Part*" button to perform wavelet denoising of the selected variables. Clicking the "*Cancel*" button will leave the dialogue box without performing the wavelet denoising operation;

Step 3.7: Optionally, select the "*Batch Print ...*" menu item within the "*File*" pull-down main menu to print one or more graphs of denoised signals.

Procedure 4. Apply the denoising feature to all signals in the selected input SEDE file by clicking the "*Build*" item within the "*File*" pull-down menu:

Step 4.1: From the PAW main menu, click the "*File*" item to show a pull-down menu, and then select the "*Build*" item to activate a BUILD dialogue box;

Step 4.2: Select a norm file for "*Previous Norm File*". Usually, a previous norm file can be found in a directory like "*c:\paw\longterm*". The default file is satisfactory for use with Wavelets;

Step 4.3: Click the "*Wavelet*" button within MODELS block of the dialogue box;

Step 4.4: Enable the "*Over Write*" and "*Global Removal*" options as if necessary by clicking the options. Both options are enabled as default;

Step 4.5: Click the "*Build All*" button to perform wavelet denoising of all signals. Clicking the "*Cancel*" button will leave the dialogue box without performing the wavelet denoising operation;

Step 4.6: Optionally, select the "*Batch Print ...*" menu item within the "*File*" pull-down main menu to print one or more graphs of denoised signals.

Procedure 5. Once the denoised signals have been obtained by one of these four procedures they may be exported to a file of SEDE format:

Step 5.1 Perform one of the four procedures described in the above. Usually, Procedure 4 is recommended;

Step 5.2 Click the "*SEDE Format*" within the "*Output*" pull-down menu. The exported file name has the extension of "*.rst*".

Since the wavelet decomposition can be only applied to a set of data whose length is equal to 2^n, where n is natural integer from 1, 2, ..., it is suggested that the number of samples should satisfy the condition especially when only a small number of samples will be used. Otherwise, in some cases, a deviation at the end of the sampling period may occur. Another way to avoid the deviation is to change the threshold value of wavelet denoising so that more high frequency components will be retained in the denoised signals.

The maximum number of signal samples to be denoised in PAW is 2048, which will meet most applications. However, when the number of samples of selected signals is larger than 2048, PAW will produce quality denoised signals. In few cases where consistent jumps exist in all denoised signals at the 2048th sample instance, the maximum value should be increased by a factor of 2^n with n=1, 2, ... In this case, some special technique is needed in order to produce quality denoising results.

4.2 Results

Results depicted in Figures 5-10 demonstrate significant advantages of the wavelet technology over traditional filtering technologies. Three contaminated signals and their denoised counterparts are shown in Figures 5 and 6, respectively. Comparison of denoised signals with their raw data is presented in Figures 7 and 8. The results in the figures clearly shown that the denoised signals contain major dynamics of the raw data and only the noise components are removed. Use of the wavelet denoising technology during the start-up operation at PLGS is illustrated in Figures 9 and 10 where a set of raw data and denoised signals are presented.

5. CONCLUSIONS

In this paper, a wavelet representation of a signal and the shrinkage technique for denoising data are described. The filtering technology is applied to real-time data of a nuclear power plant, Point Lepreau Generating Station (PLGS). In particular, this technology has been built into our Windows-based software package, Plant Analysis Workbench, and become a routine operation in PLGS. It has been shown that the filtering technology is very powerful in removing noise from contaminated signals without specifying the frequency band-width of the signals. It also works very well when the signals' frequencies change within a time period under consideration. In fact, the developed Wavelet-based filtering technology has been used successfully during recent start-up operations at PLGS. The convincing results will stimulate more research in the wavelet-based approach to signal processing. Given a poor signal-to-noise ratio, more advanced thresholding technique should be used. The approach described in this paper can handle the situation where noise levels of signals are different.

6. REFERENCES

[1] Hallat, S. "A theory for multiresolution signal decomposition: the wavelet representation," IEEE Trans. on PAMI, Vol. 11, p.674, 1989.

[2] Daubechies, I. "Ten Lectures on Wavelets," SIAM Press, 1992.

[3] Lina, J.M. and M. Mayrand, "Complex Daubechies wavelets," Applied and Computational Harmonic Analysis, Vol. 2, pp. 219, 1995.

[4] Donoho, D. and I. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," Biometrika, Vol. 81, pp. 425, 1994.

[5] Donoha, D., "De-noising by soft-thresholding," IEEE Trans. on Information Theory, Vol. 41, pp.613, 1995.

[6] Thompson, P.D., M.K. Gay, C. Xian and J.W. Thompson, "The Plant Analysis Workbench, A tool to assist in plant data monitoring and diagnostics," International Nuclear Congress 93, October, 1993.

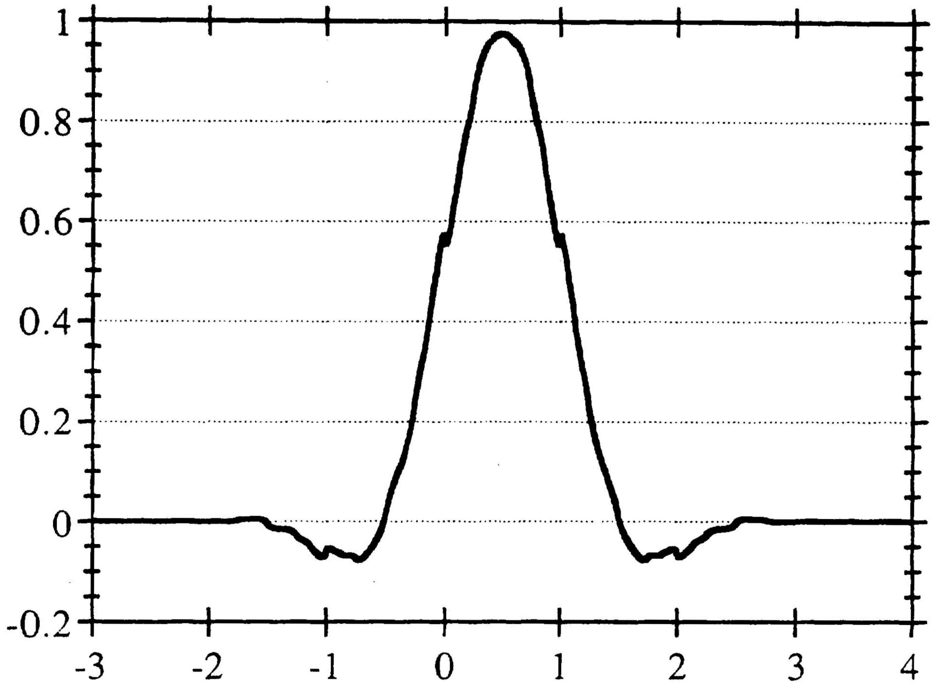

Fig.1 Real part of the scaling function

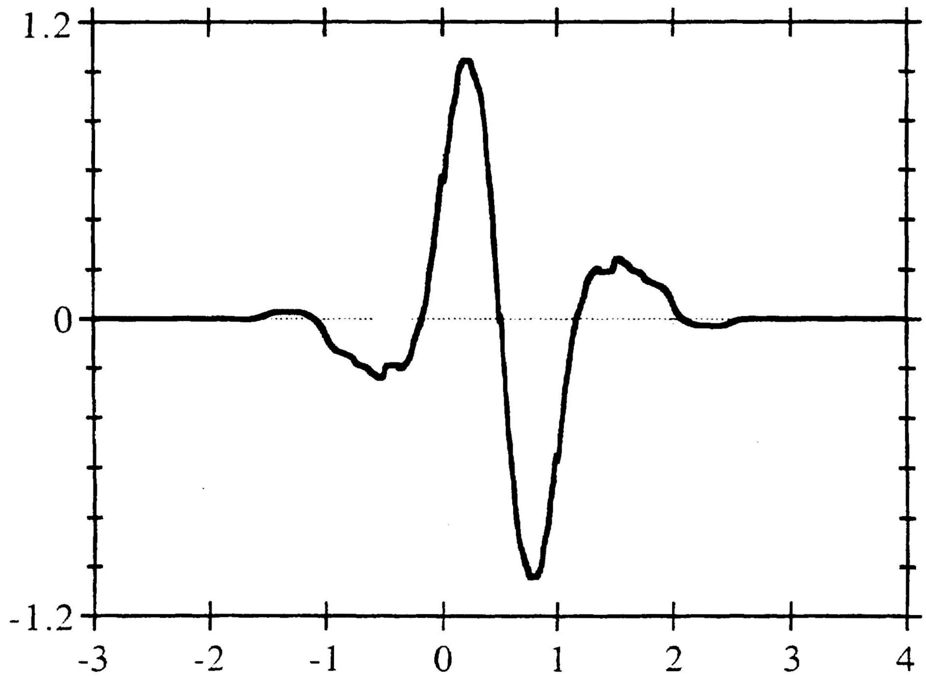

Fig.2 Real part of the wavelet

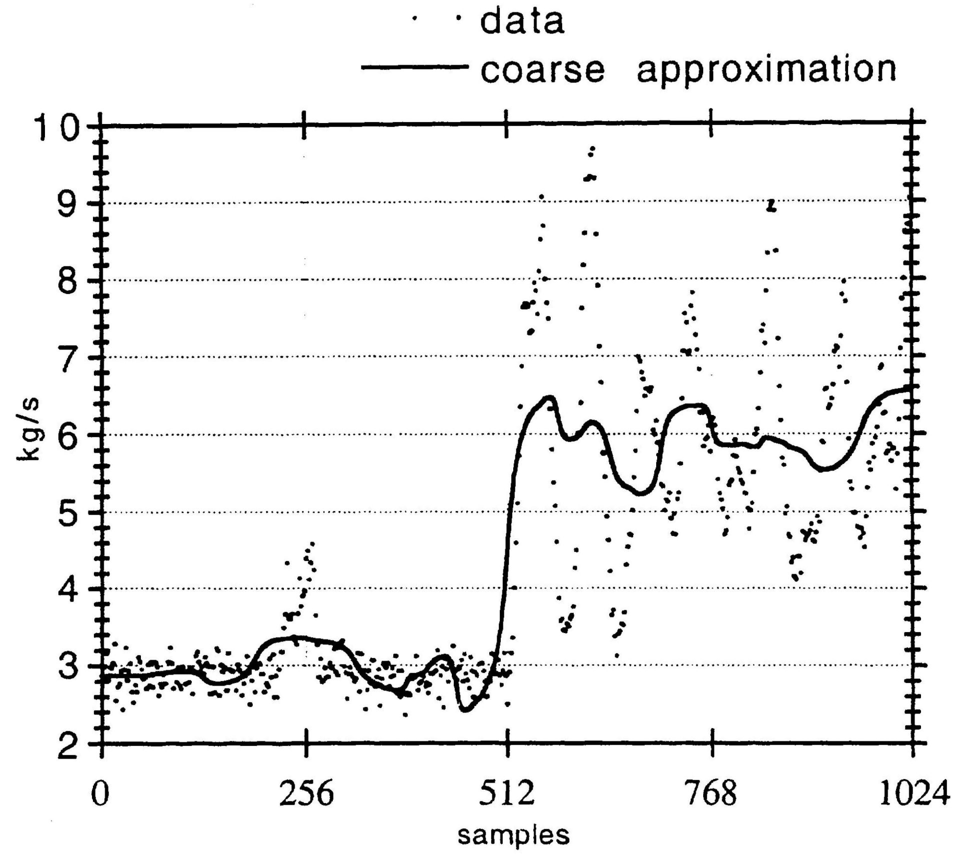

Fig.3 original data (HT BL Flow) and low resolution approximation

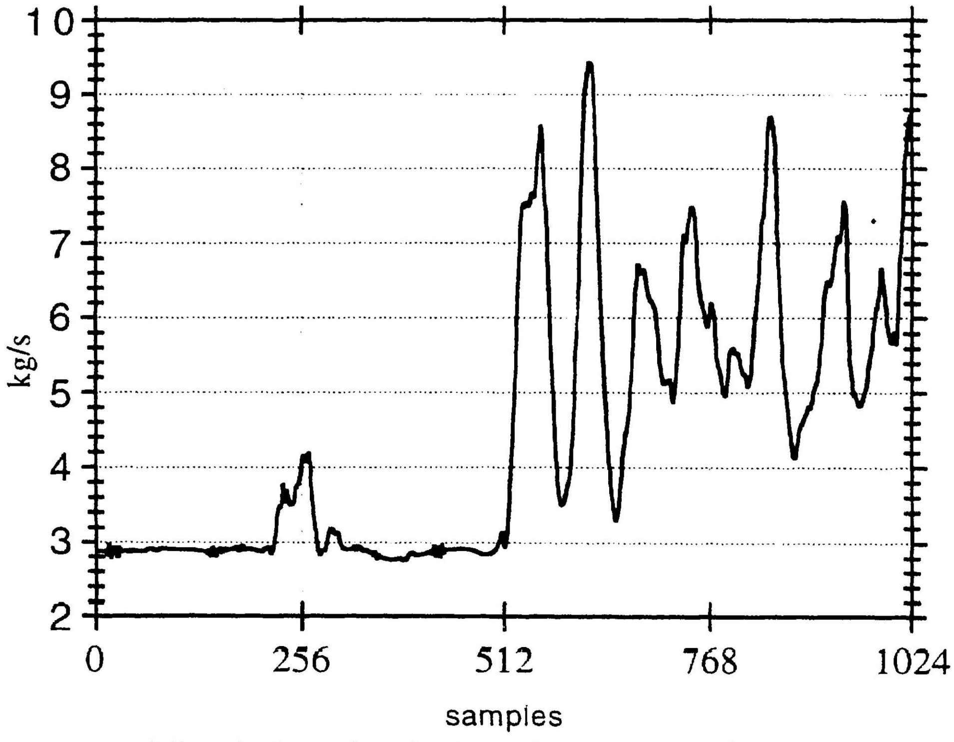

Fig.4 denoised signal (data on fig.3)

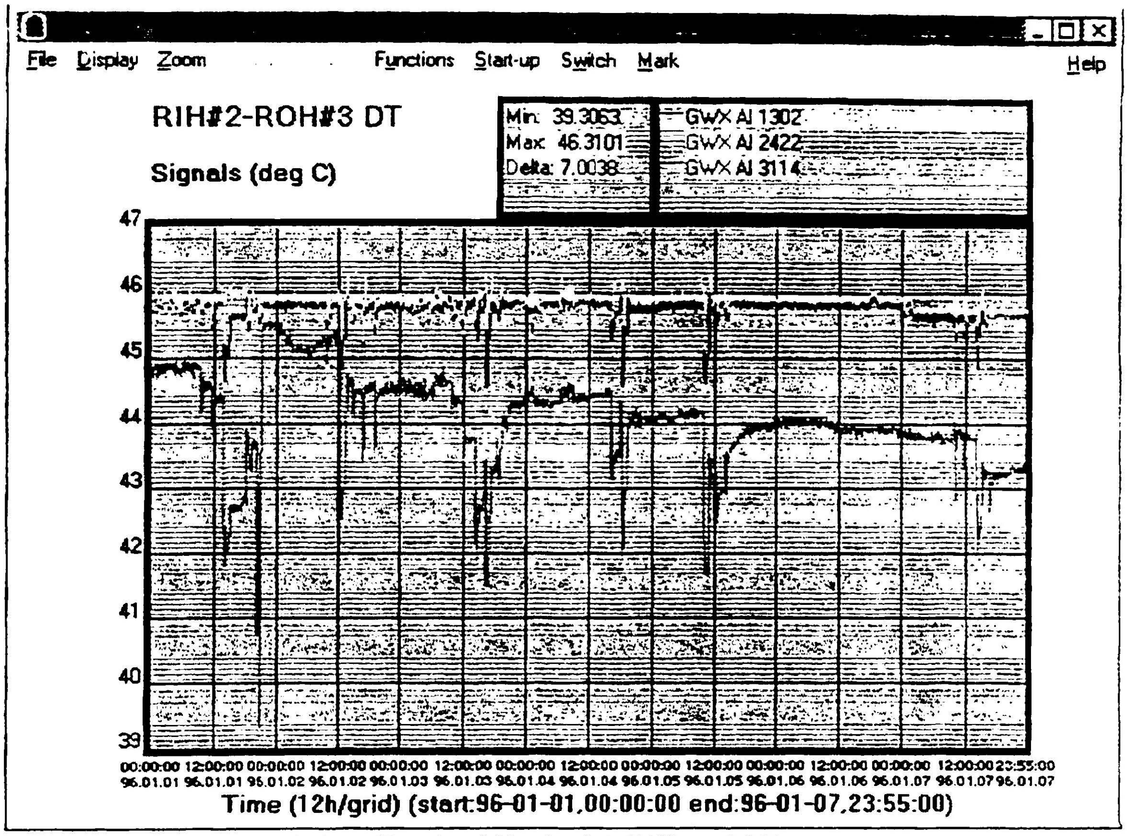

Fig. 5 Raw Signals of Header 2-3 Differential Temperature

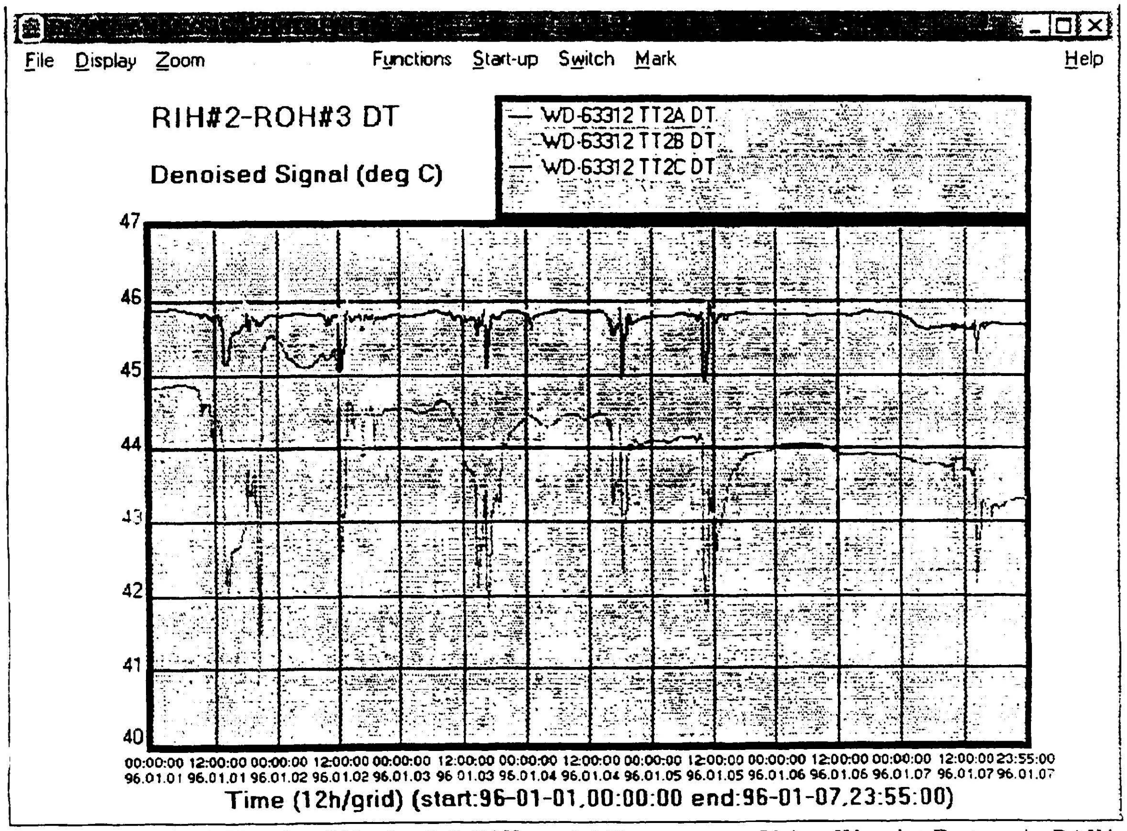

Fig. 6 Denoised Signals of Header 2-3 Differential Temperature, Using Wavelet Feature in PAW

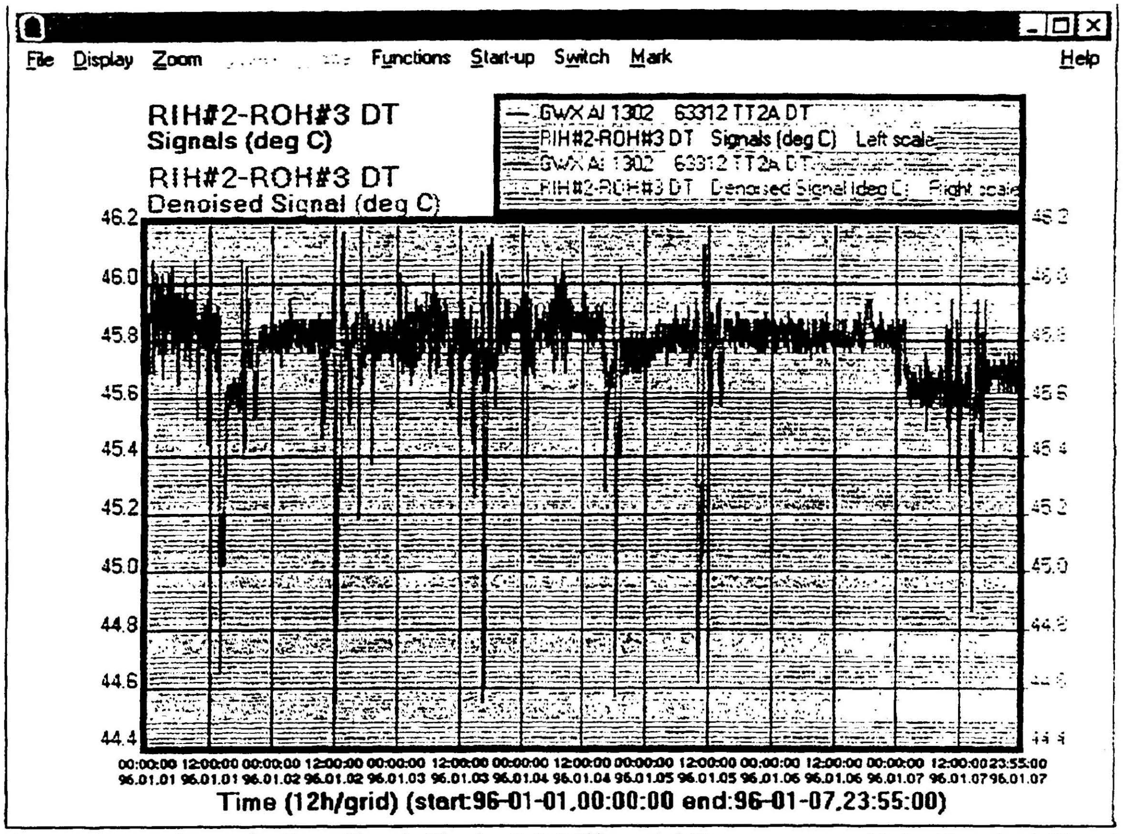

Fig. 7 Comparison of a Raw Signal, Header 2-3 Differential Temperature, with its Denoised Signal

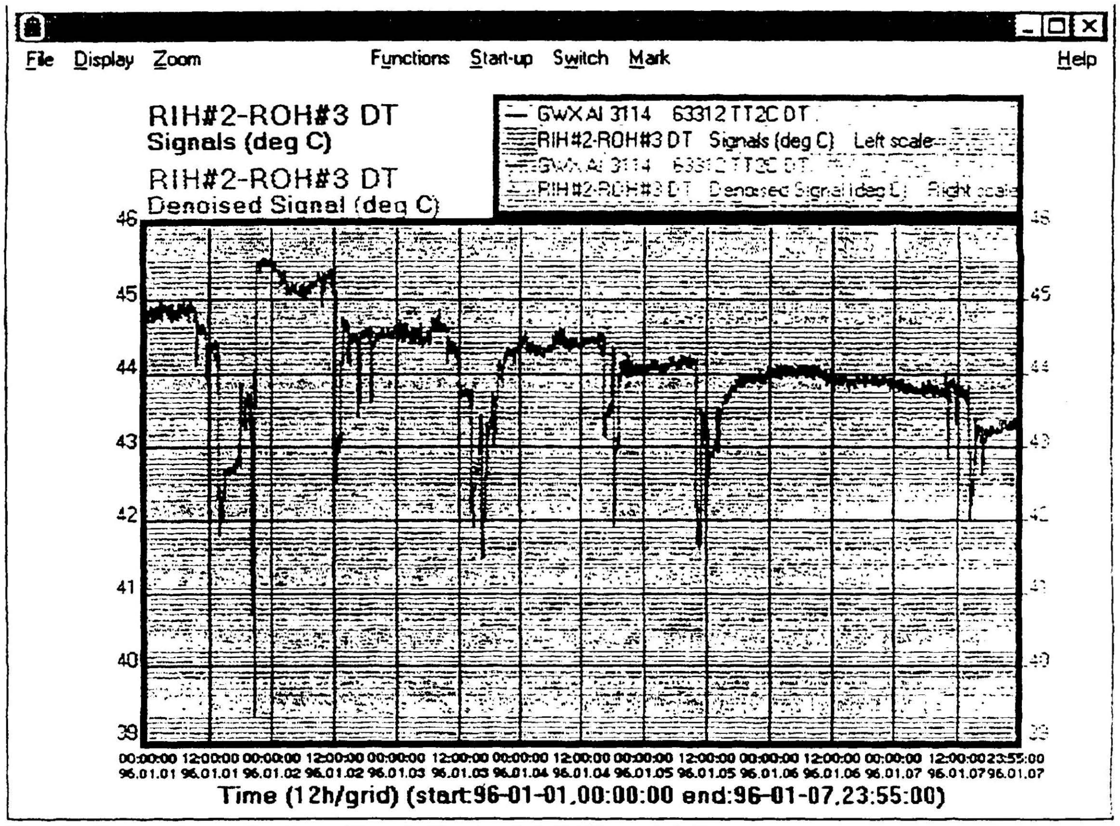

Fig. 8 Comparison of a Raw Signal, Header 2-3 Differential Temperature, with its Denoised Signal

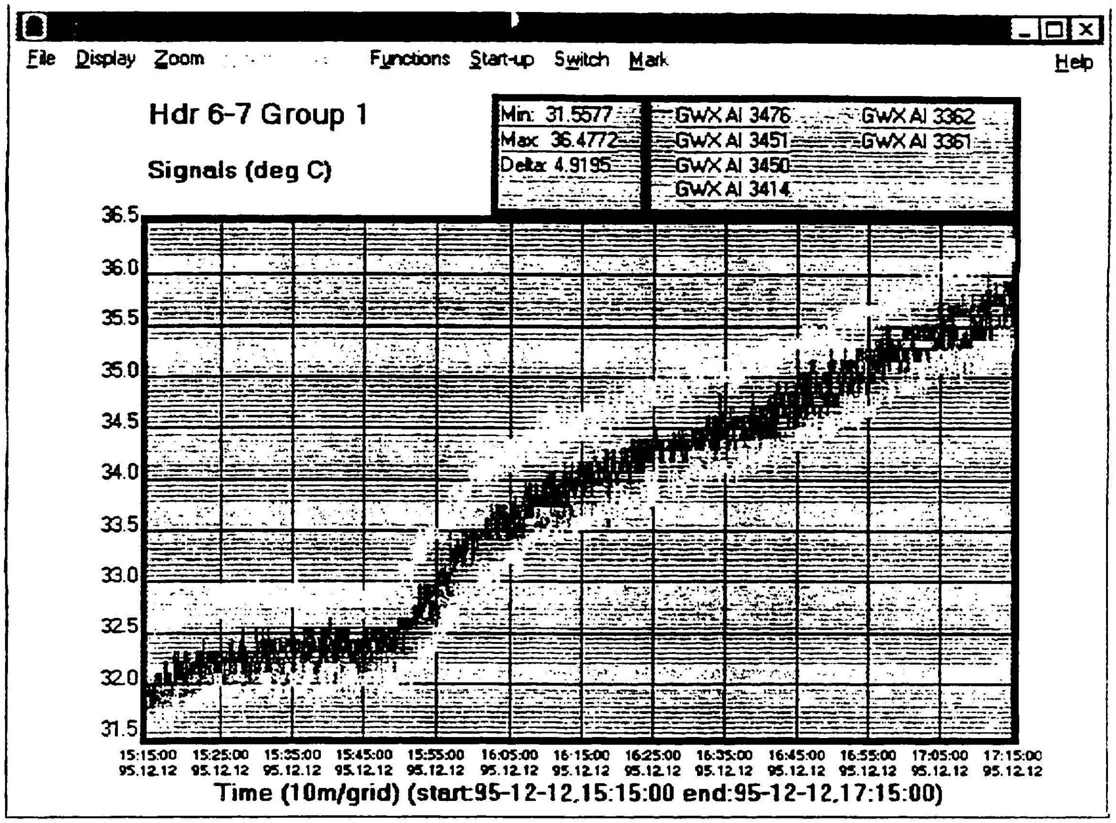

Fig. 9 Raw Signals of Channel Temperature During Start-up Operations in PLGS

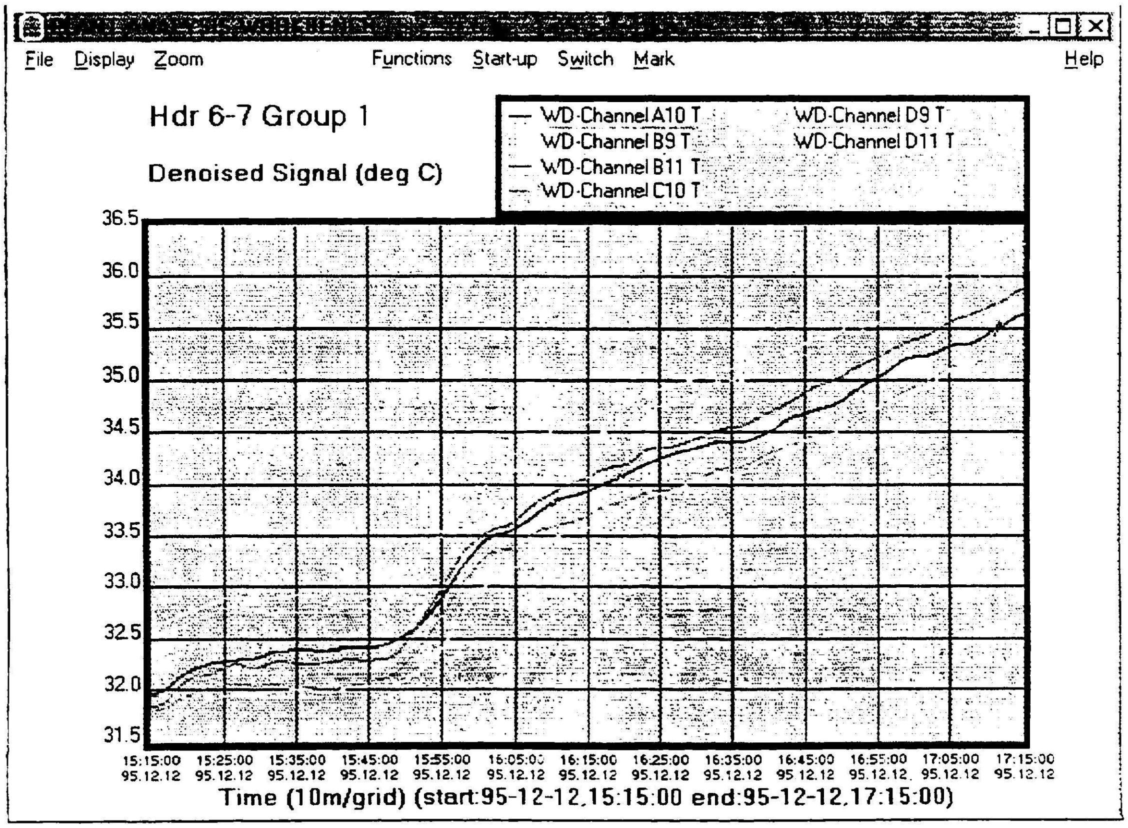

Fig. 10 Denoised Signals of Channel Temperature During Start-up Operations, Using Wavelet in PAW