Selected applications of wavelet analysis in signal processing

Janusz Walczak, Michał Lewandowski
Silesian Technical University, Electrical Engineering Department, Institute of Electrotechnics and Informatics,
44-100 Gliwice, ul. Akademicka 10, email: janusz.walczak@polsl.pl, michal.lewandowski@polsl.pl

Abstract – In this paper an application of a wavelet transform to a pulse interference detection and a noise reduction was described. The proposed algorithms are based on a discrete wavelet representation of an analysed signal. To illustrate the results some examples are provided.

I. INTRODUCTION.

Due to an increasing complexity of the problems which are solved using a DSP techniques, the time-frequency signal representations are becoming more popular every day. The family of a wavelet transforms is a good example of such a representation. The most important advantages of a wavelet transform are coming from its two main features [4], [10]. First is a natural ability to show a signal in a time and scale domain simultaneously. It lets one to analyse nonstationery signals. The second is the ability to show the signal with different scales. It lets one to evaluate the signal low and high frequency components from the same analysis with a quite good resolution [5]. Another important feature of the wavelet transform is a wide range of available wavelet bases [7], [10]. Because the requirements for a wavelet basis are well known [4], it is even possible to construct a new basis [4], [10]. The wavelet bases may strongly differ in their properties. It gives the possibility to chose the best basis for a certain application. Mentioned properties of the wavelet transform are the source of its great flexibility. It can be used for solving a broad range of DSP problems like noise reduction, signal compression, identification of signal anomalies and many more [2], [7]. In this paper the discrete wavelet transform is used in algorithms for pulse interference detection and noise cancellation of one dimensional signals.

II. THE WAVELET REPRESENTATIONS OF THE SIGNALS.

Among many different time-frequency representations of the signals, the wavelet transforms are quite popular. Some of them are described below.

A. *Continuous representations* .

Continuous wavelet transform (CWT) of a $f(t) \in L^2$ signal, can be defined as follows [1]:

$$CWT(t',s) = \int_{-\infty}^{\infty} f(t) \cdot w^*(t',s)dt, \quad s \in R^+, t' \in R, \qquad (1)$$

where:

s- scale variable,
t'- translation variable,
CWT(t',s)-CWT of a signal f(t),
*operator of conjugation,
f(t)- analysed signal,
w(t',s)-wavelet basis.

The right side of the equation (1) is a product of signal *f(t)* and wavelet function *w(t',s)* for a certain scale *s*, and translation *t'*. The *s* parameter, roughly speaking, can be interpreted as an inverse of a signal frequency [1], [10]. The wavelet function *w(t',s)* for a certain scale and translation can be generated from a mother wavelet using the following formula [1]:

$$w(t',s) = \frac{1}{\sqrt{s}} w'\left(\frac{t-t'}{s}\right), \quad s>0, \qquad (2)$$

where:

$w'(\cdot)$ -mother wavelet.

A recovery of the signal from the CWT coefficients can be done by an inverse continuous wavelet transform (ICWT) which can be defined as follows [1]:

$$f(t) = \frac{1}{c} \int_s \int_{t'} CWT(t',s) \cdot w\left(\frac{t-t'}{s}\right) \cdot \frac{dt'\,ds}{s^2}, \qquad (3)$$

where:

f (t)-recovered signal,
CWT(t',s)-CWT of a f(t) signal,
c-constant of the admissibility condition.

The quantization of the *s* and *t'* parameters in the equation (2), gives the following result:

$$w_{m,n}(t) = s_0^{-m/2} \cdot w(s_0^{-m} \cdot t - n \cdot t_0'), \quad m,n \in Z^2 \quad . \qquad (4)$$

There is a parameter set: s_0=2 and $t_0' = 1$ very popular in practical applications. It is so popular, because it allows to reduce the wavelet transform redundancy. This set is called a dyadic grid [1].

B. *Discrete representations*.

The DSP operates only on the discrete signals, thus it is compulsory to use signal transforms in their discrete versions.
The discrete wavelet transform (DWT) can be defined as follows [1]:

Publishing info:
Walczak J., Lewandowski M.: "Selected applications of wavelet analysis in signal processing", Proceedings of the IX ZKwE Conference, Poznań, April 2004, pp. 21-24.
(If you like it please make a citation!)

$$DWT_{m,n} = \sum_i x[i] \cdot w_{m,n}(i), \qquad (5)$$

where:

$DWT_{j,k}$ – DWT of a discrete signal x[i],
x[i] – discrete signal,
$w_{m,n}$ – wavelet with scale index m, and translation index n.

The inverse discrete wavelet transform (IDWT) can be defined as follows [1]:

$$x(i) = \sum_m \sum_n DWT_{m,n} \cdot w_{m,n}, \qquad (6)$$

where:

$DWT_{m,n}$ – DWT of the signal x[i],
x[i] – the discrete signal.

It can be proven [6], [10] that there is a direct relation between DWT and the subband coding scheme (in this version called the Mallat algorithm). This relation shows, that it is possible to directly calculate coefficients of DWT using the Mallat algorithm. However, it is possible only if one can assign a QMF (quadrature mirror filter) bank to the used wavelet basis. Fortunately there are lots of wavelet bases which obey this condition [2], [7]. In the Mallat algorithm, two FIR filters are used on each filtration stage (these filters are marked as h[k] and g[k] in figure 1 and 2). These filters are directly related to the chosen wavelet basis. The results of the filtration done by g[k] and h[k] are respectively called a detail and an approximation (fig. 1) [1]. The detail contains the wavelet transform coefficients at a certain scale level. The approximation can be filtered again using the same scheme to obtain wavelet transform coefficients at a scale two times higher, four times higher and so on (fig. 2). The change of the scale is made by the downsampling operation shown in the fig. 1 and 2. Fig. 2 shows a multistage signal decomposition performed by the Mallat algorithm.

The procedure described above is often called the fast wavelet transform (FWT). It is a very popular application of a wavelet transform on a DSP ground, due to it's computational efficiency and numerical stability.

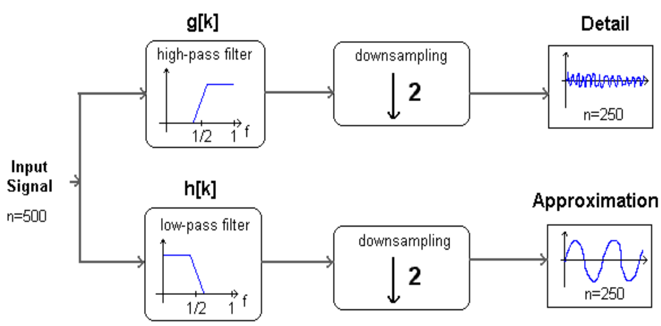


Fig.1. One level FWT on a dyadic grid.

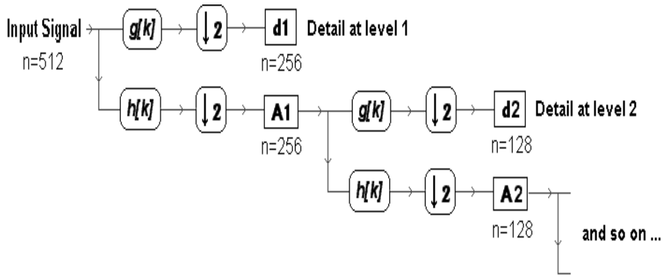


Fig.2. Multilevel FWT on a dyadic grid.

Inverse fast wavelet transform (IFWT) can be done using an algorithm, which is exactly reverse to FWT (fig. 3) [1]. In IFWT $g^{-1}[k]$, $h^{-1}[k]$ filters are used. These filters are a mirror filters to g[k], h[k] respectively. To reverse the scale changes made by downsampling in FWT, in IFWT the upsampling process is used (see fig. 3).

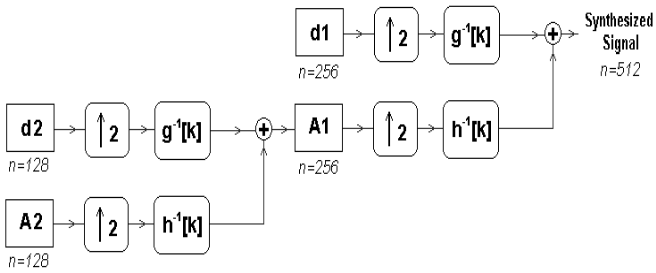


Fig.3. Multilevel IFWT.

III. THE ALGORITHMS OF WAVELET FILTERING.

Currently, there are many available engineering tools with a wavelet analysis options. One of the most well known is the "Wavelet Toolbox" of the Matlab package [7]. This toolbox has an impressive set of functions for both DWT and FWT and a GUI tool for comfortable usage. In spite of it's broad range of implemented functions, the "Wavelet Toolbox" has some serious limitations when used for DSP. The main of them are:

- Quite low computational efficiency coming from the Matlab's internal architecture (using of an interpreted language, high requirement of Matlab's package itself). The fact, that most of the functions from wavelet toolbox are rather general purpose functions, which are not optimised for efficiency (like Matlab's FFT routines for example), might be also significant.
- Keeping all of the intermediate computation results in RAM. The bigger the analysed signal is, the more system's operational memory is needed.
- There is a lack of details about the wavelet algorithms implementation in „Wavelet Tollbox" documentation.

Because of these reasons it is hard to keep a direct control over the camputational process. Especially it is hard to

Publishing info:
Walczak J., Lewandowski M.: "Selected applications of wavelet analysis in signal processing", Proceedings of the IX ZKwE Conference, Poznań, April 2004, pp. 21-24.
(If you like it please make a citation!)

evaluate the efficiency of the used algorithms and do some changes in the toolbox internal functions (because of the dependencies between the functions). To avoid the problems described above, it was decided to implement the wavelet algorithms directly in C++ programming language. It gives a full control over the implementation details. Another advantage of C++ is a fact, that this language is a standard in DSP programming nowadays, so it should be easy to adopt the code for a DSP processor.

A. *Pulse interference localization*.

Due to the fact, that wavelet transform gives a possibility to precisely locate changes of a wavelet spectrum in time, it could be used to locate moments of the abrupt changes in the signal [1]. An example of such a change might be a pulse interference. An algorithm for a localization of such interferences is proposed below. The algorithm is based on a assumption that coefficients of an FWT detail on the first decomposition level, have a significantly bigger value in the moments when the interferences occur [7]. If this assumption is true, one can localize the exact moments of the interference occurrence using a thresholding method. The algorithm is proposed as follows:

1. Performing a one level FWT of the signal to obtain the first level detail.
2. Analysis of the obtained detail to determine the detection threshold. One of the simplest methods to do that, can be an evaluation of an absolute mean of the detail and then multiplication by a constant (correction ratio) to avoid false alarms (eq. 7).

$$ths = c \cdot \frac{1}{n} \cdot \sum_{i=1}^{n} abs(d[i]), \qquad (7)$$

where:
ths – detection threshold,
d[*i*] – *i*-th detail coefficient,
abs() – absolute value operator,
n – number of detail coefficients,
c – correction ratio.

3. Scanning through all detail coefficients and saving the value of those which meet: $d[i] > ths$.
4. Multiplication of each value in vector *d* by a factor of 2^s, where *s* is a scale index (in this case equal 1). It is necessary to even out the scale of the detail and the analysed signal.

The vector *d* obtained this way should contain indices of these samples in the analysed signal which are interfered by a pulse signal.

B. *Noise reduction*.

The proposed algorithm [8] is based on an assumption, that noise, as a random signal, is much less correlated with the wavelet basis than the useful signal hidden in the noise. If this assumption is true, the wavelet coefficients connected with the noise should be significantly smaller than those connected with the useful signal. Following this rule a method of noise reduction can be proposed. This method uses thresholding to distinguish the useful signal from the noise and cancel the latter. All wavelet coefficients smaller than a fixed threshold are discarded (set to zero). When a signal is reconstructed after such a thresholding, it should contain much less noise then before this procedure. The procedure is proposed as follows:

1. FWT of a selected signal.
2. Evaluation of the threshold for details at each computed scale.
3. Thresholding of the details using an evaluated threshold.
4. IFWT to obtain reconstructed signal.

Using the algorithm described above one can expect, that the final result will be better when the correlation between the wavelet and the useful signal will be stronger. On the other hand, the correlation between the noise and the wavelet should be as low as possible [7]. This leads to conclusion, that the choice of the wavelet basis is very important for this algorithm. Another important thing is the method used to evaluate the threshold.

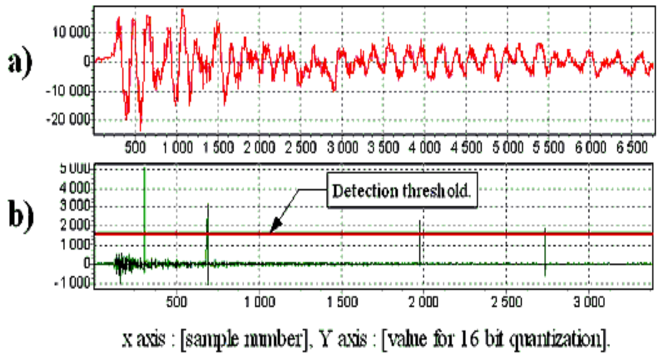
IV. Examples of applications.

Examples presented below are the applications of the algorithms proposed in the previous paragraph. These algorithms were tested on selected signals.

A. *Pulse interference localization.*

To verify the effectiveness of the proposed algorithm, a nonstationary signal was used. First, a short sample of a string vibration of a musical instrument was recorded. Next, the four pulse interferences were added to the signal. Signal with added interferences is shown in fig. 4a. In the next step, the pulse interference localization algorithm was performed on the prepared signal. In fig. 4b the detail obtained by FWT can be seen. For the FWT the *Daubechies 5* [4] wavelet was used. As can be seen in figure 4b, the algorithm revealed all four interferences (they all are bigger than the detection threshold).

Publishing info:
Walczak J., Lewandowski M.: “Selected applications of wavelet analysis in signal processing”, Proceedings of the IX ZKwE Conference, Poznań, April 2004, pp. 21-24.
(If you like it please make a citation!)



Rys.4. a)Nonstationery signal with four pulse interferences. b) First level detail for detection threshold with c= 20.

B. *Noise reduction.*

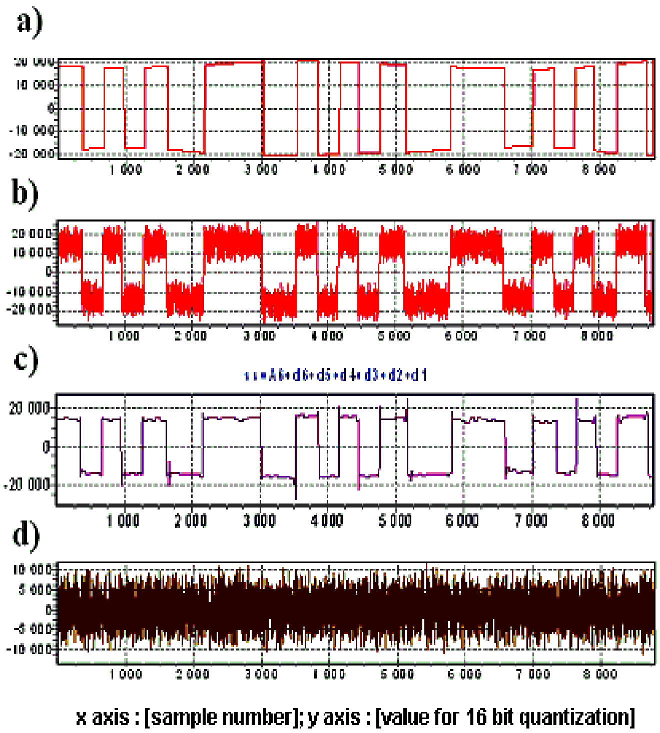
To show the effectiveness of the noise reduction algorithm, a modulated square wave and white noise was used. The modulated square wave (an amplitude modulation was used) was chosen as a test signal, because of its broad frequency spectrum. The modulated square wave is shown in fig. 5a. In the figure 5b one can see a test signal mixed with white noise. In the next step, the noise reduction algorithm was performed on the signal from fig. 5b. A six stage FWT with the Haar wavelet was performed. The detection threshold was computed using the formula shown below:

$$thn_j = k \cdot \sqrt{\frac{\sum_{i=1}^{n_j} \left(d_j[i]\right)^2}{n_j}}, \qquad (8)$$

where:

thn_j – nose threshold for j-th detail,
$d_j[i]$ – *i*-th coefficient of a j-th detail,
n_j – number of coefficients of a j-th detail,
k – correction ratio (equal 2 is this case).

Formula (8) was only used to set the thresholds roughly. In the next step threshold for each detail was corrected by hand. After that, IFWT was applied to obtain the synthesized signal. Effect of the synthesis is shown in fig. 5c. In fig. 5d the difference between 5c and 5b is shown. As can be seen this difference contains mainly noise.



Rys.5. a) Modulated square wave. b)Square wave with white noise. c)Result of the noise reduction algorithm. d) Difference between b) and c).

V. SUMMARY.

Two algorithms based on some features of the wavelet transform were presented in this paper. First is an algorithm for pulse interference detection and localization and second is an algorithm for noise reduction. It both cases the fast wavelet transform (FWT) were used. The algorithms were implemented in C++ programming language and tested on some selected signals. Simulation tests confirmed the assumptions which were made as a base of the algorithms. During the C++ implementation process some problems, which can be particularly important for DSP, were revealed:

- Due to the fact, that the number of FWT coefficients increases with the increase of the analysis level, the algorithm can consume significant amount of system's memory.
- The choice of the method of threshold calculation (in both algorithms) has significant influence on the final results.
- Because the FWT algorithm uses a cascade of FIR filters, it introduces a delay between input and output signal. This delay is proportional to the filter length and analysis level and can be a serious limitation in some applications (for example : real time voice compression).

Because problems described above strongly depend on the algorithm internal features, like used analysis level, filter length etc., their influence should be considered for each

Publishing info:
Walczak J., Lewandowski M.: "Selected applications of wavelet analysis in signal processing", Proceedings of the IX ZKwE Conference, Poznań, April 2004, pp. 21-24.
(If you like it please make a citation!)

certain application. It should helps one to avoid the described problems.

LITERATURE.

[1] Białasiewicz J.: Wavelets and approximations. WNT, Warsaw, 2000. (polish)

[2] Chui C.K.: Wavelets, a tutorial, theory and applications. Academic Press, Boston, 1992.

[3] Czyżewski A.: A digital sound. EXIT Press, Warsaw, 2000. (polish)

[4] Daubechies I.: Ten lectures on wavelets. Society for industrial and applied mathematics, Philadelphia, 1992.

[5] Graps A.: An introduction to wavelets. IEEE Computational Sciences and Engineering, Vol. 2, Number 2, pp. 50-61.

[6] Mallat S. : A theory for multiresolution signal decomposition, the wavelet representations. IEEE Trans. On Pattern Analysis PAM-31, 1989, pp. 674-693

[7] Misiti M, Misiti Y, Oppenheim G, Poggi J.: Wavelet toolbox. The MathWorks, Inc, 1998.

[8] Lewandowski M.: Wavelet filtering – a B.Sc. diploma. Silesian Technical University, Gliwice, 2002. (polish)

[9] Lyons R. G.: Understanding digital signal processing. WKŁ, Warsaw, 2000.

[10] Paularikas A.D.: The transforms and applications handbook. CRC Press Inc, Florida, 2000.

Publishing info:

Walczak J., Lewandowski M.: "Selected applications of wavelet analysis in signal processing", Proceedings of the IX ZKwE Conference, Poznań, April 2004, pp. 21-24.

(If you like it please make a citation!)