

DYNAMIC RECONFIGURABLE LIFTING-BASED WAVELET PACKET PIPELINE PROCESSOR FOR REAL-TIME AUDIO APPLICATION

Alexey Petrovsky, Maxim Rodionov, Alexander Petrovsky

Department of Computer Engineering, Belarusian State University of Informatics and Radioelectronics
6, P. Brovkiy str., 220013, Minsk, Belarus
phone: + (375-17) 293-23-40, fax: + (375-17) 331-09-14, email: palex@bsuir.by
web: www.bsuir.by

ABSTRACT

In this paper, dynamic algorithm transforms (DAT) for reconfigurable real-time processor for audio application based on the adaptive wavelet packet (WP) decomposition are presented. DAT techniques is to constrain a minimum cost sub-band decomposition of wavelet transform by maximizing the minimum masking threshold (which is limited by the perceptual entropy) in every sub-band for the given embedded processor architecture and temporal resolution. The processor architecture is based on the implementation of the wavelet transform by means of its factoring into lifting steps. Practical reconfiguration strategies for the given processor are presented.

1. INTRODUCTION

The wavelet packet (WP) as a generalization of the standard wavelet transform provides a more flexible choices for time-frequency (time-scale) representation of signals [1]. In many applications, such as the design of cost-effective real-time multimedia systems and high quality audio transmission and storage. In parallel to the definition of the ISO/MPEG standards, several audio coding algorithms have been proposed that use the wavelet transform, in particular, adaptive wavelet packet transform, as the tool to decompose the signal [2,3]. In practice, WP are often implemented using a tree-structured filter bank [2-4]. The WP is a set of transformations that admits any type of tree-structured filter bank, that provides a different time-frequency tiling map. Applying the lifting scheme [5-7] for the construction of wavelets filter bank allows significantly reduce the number of arithmetic operations that are necessary to compute the transform.

Algorithm transformation techniques [8,9] such as have been employed in high-speed DSP system design. All of the above mentioned techniques are applied during the VLSI design phase and their implementation is time invariant. Therefore, this class of signal processing techniques is referred as static techniques.

Recently, dynamic techniques both of the circuit level and algorithmic level have been proposed. These techniques are based on the principles that the input signal is usually nonstationary, and hence, it is better (from a coding perspective) to adapt the algorithm and architecture to the input signal. Such systems are referred to as reconfigurable signal processing systems [10,11]. The key goal of these techniques is to im-

prove the algorithm performance by exploiting variability in the data and channel.

Our approach is to design of dynamic algorithm transform (DAT) for design of application-specific reconfigurable lifting-based WP pipeline processor, in particular, for audio processing in real-time [8,12]. The principle behind DAT techniques is to define parameter of input audio signals (subband entropy) and output encoded sequences (subband rate) for the given embedded processor architecture. Adaptive wavelet analysis for audio signal processing purposes is particularly interesting if the psychoacoustic information is considered in the WP decomposition scale. Due to the lack of selectivity of wavelet filter banks, the psychoacoustic information is computed in the wavelet domain.

2. DYNAMIC TRANSFORMATION OF WP DECOMPOSITION

We present adaptive WP tree derived via DAT's. The principle behind DAT is to define parameter of input signals (subband entropy) and output sequences (subband rate) for the given embedded processor architecture. In other hands, DAT techniques is to construct a minimum cost subband decomposition of WP by maximizing the minimum masking threshold (which is limited by the perceptual entropy (PE)) in every subband for the given computational complexity (for the given embedded processor architecture) C and temporal resolution. Achieving this purpose, we suppose that the tree structure of WP decomposition is adapted, as closely as possible, to the critical bands (CB – WPD: $(l, n) \in E_{CB}$) as shown in [8]. For the WP tree structure E_i the information density H belong to tree E_i is estimated as

$$H_{E_i} = \sum_{\forall (l,n) \in E_i} \sum_k w_{E_i}(k) \cdot \ln(w_{E_i}(k)), \quad (1)$$

where

$$w_{E_i}(k) = \frac{|x_{l,n,k}|}{\sum_{\forall (l,n) \in E_i} |x_{l,n,k}|}. \quad (2)$$

Here $x_{l,n,k}$ are wavelet coefficients, l is a decomposition level, n is the node number of decomposition level, k is the index of the current wavelet coefficient of the node (l, n) .

The growing decision for WP tree based on the given H is being taken in terms of allowing the further decomposition of the WP tree can be expressed as:

$$H_{E_i} > H_{E_{i-1}}. \quad (3)$$

If (3) is true we continue the subband splitting process in WP tree, otherwise the suboptimal decomposition for the given frame of signal is founded.

The subband splitting process is managed based on the estimated values of PE in parent and child nodes of current WP tree structure. PE estimation is described in [13,14] and expressed as

$$PE_{l,n} = \sum_{k=0}^{K_{l,n}-1} \log_2(2[n \text{ int}(SMR_{l,n,k})] + 1), \quad (4)$$

where $SMR_{l,n,k}$ is a ration between the absolute value of the wavelet coefficients $x_{l,n,k}$ in a subband of tree E_i (node (l,n)), and the corresponding masking threshold $T_{l,n}$, which is linearly spread among the $K_{l,n}$ coefficients $x_{l,n,k}$, $k = \{0, K_{l,n}\}$ of node (l,n) . The large magnitude of $SMR_{l,n,k}$ determines node (l,n) significance for PE formation.

Each allowed parent node (l,n) is split on two child nodes $(l+1,2n)$ and $(l+1,2n+1)$, if and only if the sum of $PE_{l+1,2n}$ and $PE_{l+1,2n+1}$ in the child nodes less than in the current node $PE_{l,n}$, that can written as.

$$PE_{l,n} > PE_{l+1,2n} + PE_{l+1,2n+1}. \quad (5)$$

The computational resource at current WP tree level c_s is estimated, and if it is more than the maximum allowed computation resource C the WP tree decomposition is terminated.

The dynamic WP tree structure growing level by level based on the H and corresponding time-frequency tilling map are demonstrated in a figure 1.

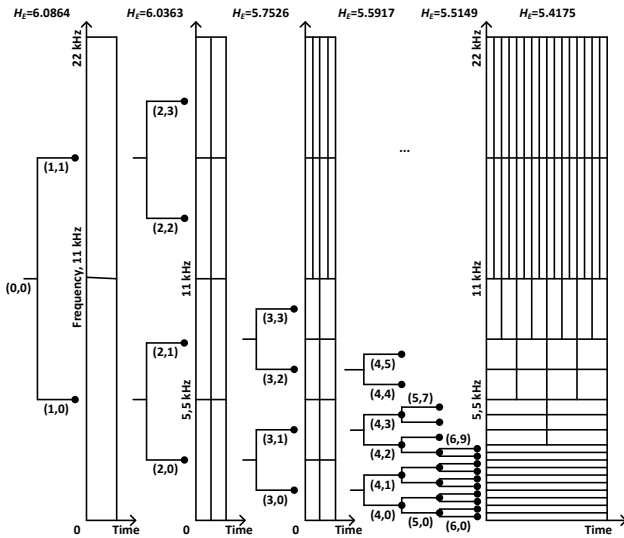


Figure 1 – WP tree structure creation and corresponding time-frequency tilling map.

Applying the information density H , the perception entropy PE , the limited WP tree structure $CB - WPD$ and the

maximum allowed computation resource C together in WP growing procedure allows us to found suboptimal solution for input signal analysis on the given hardware architecture.

3. WP IMPLEMENTATION BASED ON LIFTING SCHEME

In the tree-based scheme of the WP, each node of the tree consists of a 2-channel filter bank. Each node can be broken down into a finite sequence of simple filtering steps, which are called lifting steps or ladder structures. In [15] for two-channel filter bank proposed a method of transition from the implementation on the basis of FIR filters to architecture at the based on lifting scheme. The decomposition is essentially a factorization of the polyphase matrix of the wavelet filters into elementary matrices. As discussed in [6,15], the lifting steps scheme consists of three phases: the first step splits the data into two subsets: even and odd; the second step recalculates the coefficients (high-pass) as the failure to predict the odd set based on the even; finally the third step updates the even set using the wavelet coefficients to compute the scaling function coefficients (low-pass). This method allows to reduce on halve the number of multiplications and summations. The filtering results, finding scaling $X_{l+1,2n}(z)$ and wavelet $X_{l+1,2n+1}(z)$ coefficients relative to the input signal $X_{l,n}(z)$ in z domain can written as the follows:

$$\begin{bmatrix} X_{l+1,2n}(z) & X_{l+1,2n+1}(z) \end{bmatrix} = \begin{bmatrix} X_{l,n,e}(z) & X_{l,n,o}(z) \end{bmatrix} \times \quad (6)$$

$$\times \prod_{i=1}^{I/2} \left(\begin{bmatrix} 1 & s_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ t_i(z) & 1 \end{bmatrix} \begin{bmatrix} K_1 & 0 \\ 0 & K_2 \end{bmatrix} \right).$$

where $X_{l,n,e}(z)$ and $X_{l,n,o}(z)$ are z -representation of two sequences consisting of even and odd samples of the input signal $x_{l,n,k}$, $s_i(z)$ and $t_i(z)$ are low-order polynomials, K_1 and K_2 are constants, and I is a number of lifting steps.

Inverse decomposition of a 2-channel filter bank in the same terms can be expressed as

$$\begin{bmatrix} X_{l,n,e}(z) & X_{l,n,o}(z) \end{bmatrix} = \begin{bmatrix} X_{l+1,2n}(z) & X_{l+1,2n+1}(z) \end{bmatrix} \times \begin{bmatrix} K_1^{-1} & 0 \\ 0 & K_2^{-1} \end{bmatrix} \prod_{i=\frac{I}{2}}^1 \left(\begin{bmatrix} 1 & 0 \\ -t_i(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & -s_i(z) \\ 0 & 1 \end{bmatrix} \right). \quad (7)$$

In general case the polynomials $s_i(z)$ and $t_i(z)$ can be represented as $(b_0 + b_1 z^{-1})z^u$, where b_0 , b_1 are constants, u is the integer exponent. For example, the b_0 , b_1 and u parameters of lifting scheme for db4 wavelet mother function are presented in table 1. K_1 and K_2 are equal -0.1202 and -8.3192 correspondingly.

Table 1 – The parameters of lifting scheme for db4 (8 taps)

	$s_1(z)$	$s_2(z)$	$s_3(z)$	$t_1(z)$	$t_2(z)$
b_0	3.1029	-5.1995	0.3141	0.0763	-3.1769
b_1	0	1.6625	0	-0.2920	-0.0379
u	0	-1	-3	1	3

In this work, for fixed point WP implementation an arithmetic with an arbitrary number of integer and fractional bits is used as proposed in [16,17]. The advantage of this number representation is the fact that it can be realized using conventional integer arithmetic resource.

The block diagrams of the lifting step for the cases when $b_1 = 0$, and for the case when $b_1 \neq 0$ (see) are shown in figure 2 and figure 3 correspondingly. The solid and the dashed lines at the begin and the end of the block diagrams are associate with the instances $s_i(z)$ and $t_i(z)$ accordingly. The arithmetic shift right operation (block *shift*) is realized in compliance with the implemented fixed-point algorithm and amounts to a signal lines shift before submitting them to the adder input. So, the shifters are not used system hardware resources.

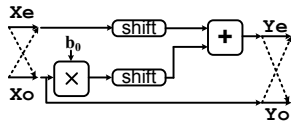


Figure 2 – Block diagram of the lifting step when $b_1 = 0$

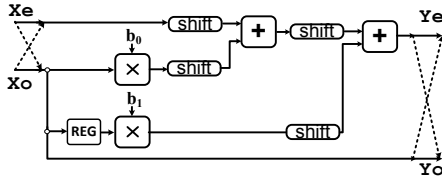


Figure 3 – Block diagram of the lifting step when $b_1 \neq 0$

4. PIPELINE PROCESSOR WITH DYNAMIC RECONFIGURABLE ARCHITECTURE

4.1 Run time reconfigurable architecture

The structure of reconfigurable DSP system for signal analysis based on DAT approach consists of the specific microprocessor oriented on the signal processing (DSP microprocessor) and WP processor itself with the reconfigurable architecture. The DSP microprocessor perform several task, such as: processing wavelet coefficients $X_{l,n,k}$ in sub-bands (l,n) that corresponds to the current WP tree structure E_i ; estimate H_{E_i} and $PE_{l,n}$; obtain the reconfiguration vector for WP processor $r_{l,n}, (l,n) \in E_i$. WP processor is realized on pipeline architecture with dynamic reconfiguration for

implementing adaptive WP. The length of the pipeline is obtained from the limited WP tree structure ($CB - WPD$).

A great dependence of the process on the WP structure grows leads to the necessity of introducing an easily reconfigurable parallel-pipeline structure with computation resource C . Thus, the DSP system for audio processing based on DAT-approach consists as shown on figure 4.

4.2 WP lifting based pipeline processor

The pipeline architecture is applied for effective implementation of the WP algorithm, as it confirmed in [7,18].

We suggest the pipeline architecture for constructing the WP lifting based processor. This architecture integrates the sequential connection of the homogeneous block that implement a two-channel filter bank that allows to calculate WP with an arbitrary tree structure. The maximum number of decomposition level that can be realize is 8, it is associate with the depth of the $CB - WPD$.

Thus, the basic decomposition of WP expressed as processing unit (PU) which acts a two-channel filter bank based on lifting scheme. The block diagram of PU is shown in figure 5. The input sequence $x_{l,n,k}$ is split into even X_e and odd X_o samples in PU before the processing is started according to the lifting scheme. The structure of PU has the following abbreviations (see figure 5): wl is a bit capacity; l is a number of elementary steps of the lifting scheme; V_{PE} is a vectors, each element of it is a set of the parameters of the same elementary step of the lifting scheme; V_{BUF} is a vector, each element of it specify the number of delays, respectively in the upper and lower channels after the same elementary step. The present elements corresponds to FIFO registers that, on the one hand are delay elements z^{-1} in the algorithm and, on the other hand, makes possible a pipelined realization in architecture for throughput performance increasing. The coefficients K_1 and K_2 applies to the result of lifting scheme as it described in (6). The estimated hardware resources required for PU implementation are shown in a table 2.

Table 2 – Estimation of hardware resources for PU implementation

Resource type	Utilized
Multipliers $wl \times wl$	$N+2$
Adders(wl – bit capacity)	N
Registers(wl – bit capacity)	$N+1$
Multiplexers 2-in-1 (wl – bit capacity)	1

N – is the number of filter taps.

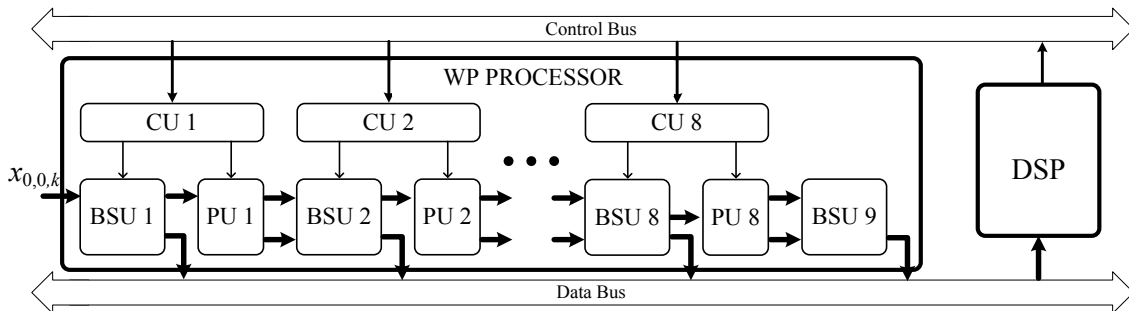


Figure 4 – DAT-based reconfigurable signal processing system.

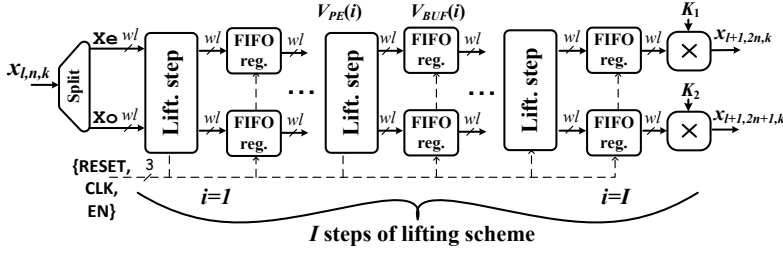


Figure 5 – Block diagram of the PU.

Next, buffer/switch unit (BSU) realizes double buffering scheme known as “ping-pong” for providing parallel access to the data for storing results and getting source data from/for PU. The additional channel is for outputting the result data. The two output streams of samples $x_{l+1,2n,k}$ and $x_{l+1,2n+1,k}$ from l -th PU are stored in BSU and simultaneously $l+1$ -th PU can get the samples for the next processing stage. Unified block diagram of BSU is represented on the figure 6. Each BSU in parallel-pipeline architecture has addressed a different memory size that depends on the WP decomposition level.

The reconfiguration vector $r_{l,n}$ decoding, memory address generation, PU enabling, data exchange controlling and the pipe-line synchronizing are performed by the control units (CU). All this functions in CU at each WP processor stages is carried out in parallel. The stages work of the pipeline WP processor are synchronized according to the DAT’s techniques.

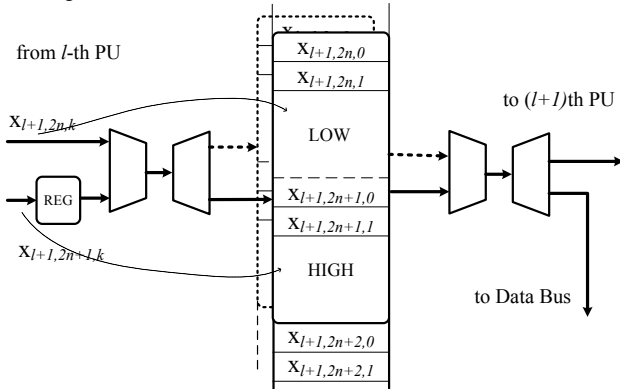


Figure 6 – Unified block diagram of the BSU.

4.3 Dynamic WP tree decomposition algorithm

Suppose, a computation resource $(m, r_{l,n})$ is limited by C . The limiting WP tree structure is $CB - WPD$: $(l, n) \in E_{CB}$. Suppose, the required computation resource of i -th WP tree $(l, n) \in E_i$ is defined as value c_i . The split decision of dividing the parent node (l, n) on the two children $(l+1, 2n)$ and $(l+1, 2n+1)$ is referred as a $split(l, n)$ where l is a transformation scale level, n is n -th node of the level l , m is a stage of the pipeline processor. Number of the input frame of the audio signal is j .

STEP 1. Let $j = 1$, $m = 1$, $l = 0$ and $split(l, n) = YES$, $r_{l,n} = YES$, WP tree root node is $(0,0)$ for the first frame

of the input audio signal with perceptual entropy $PE_{0,0}$, information density $H_{E_{0,0}}$ and reconfiguration process is allowed.

STEP 2. $i = j$, the first frame of the input audio signal defines growing process of WP tree structure. Making the signal decomposition is based on 1st PU.

STEP 3. Estimate the perceptual entropy in each node $PE_{l,n}$ and information density $H_{E_{1,1}}$ of the WP tree structure.

STEP 4. Check the WP tree structure information density $H_{E_{1,1}}$ in comparison with the WP tree $E_{0,0}$

IF $H_{E_{0,0}} < H_{E_{1,1}}$,

THEN that is not an audio signal, the coefficients are not processed and GOTO STEP 1.

STEP 5. $l = l + 1$.

IF $l - 1 > \text{maximum of the scale level of the limiting WP tree structure } CB - WPD$,

THEN STOP – the growing process for the WP tree structure $E_{i,m}$ is finished.

STEP 6. Check the WP tree structure $E_{i,m}$ nodes belonging to $CB - WPD$: E_{CB} :

IF $(l, n) \in E_{m,i} = (l, n) \in E_{CB}$, THEN $r_{l,n} = NO$.

STEP 7. $m = m + 1$. WP tree structure $E_{i,m}$ growing is performed as follows

For each node n of the level l :

- estimate and check the adequacy of the WP tree computation resource $c_{i,m}$:

IF $c_{i,m} > C$,

THEN $r_{l,n} = NO$ and STOP – the growing process for the WP tree structure $E_{i,m}$ is finished.

- perform the decomposition of the parent node (l, n) .

- calculate the perceptual entropy into the child nodes $PE_{l+1,2n}$ and $PE_{l+1,2n+1}$:

IF $PE_{l,n} \geq PE_{l+1,2n} + PE_{l+1,2n+1}$,

THEN $split(l, n) = YES$, $r_{l,n} = YES$

ELSE $split(l, n) = NO$, $r_{l,n} = NO$.

STEP 8. $j = j + 1$. Read the next frame of the input audio signal. It will be processed according to the WP tree structure $E_{i,m}$.

STEP 9. Estimate the information density $H_{E_{i,m}}$ of the WP tree structure $E_{i,m}$:

IF $H_{E_{i,m}} > H_{E_{i,m-1}}$,

THEN $r_{l,n} = NO$ and STOP – the growing process for the WP tree structure $E_{i,m}$ is finished.

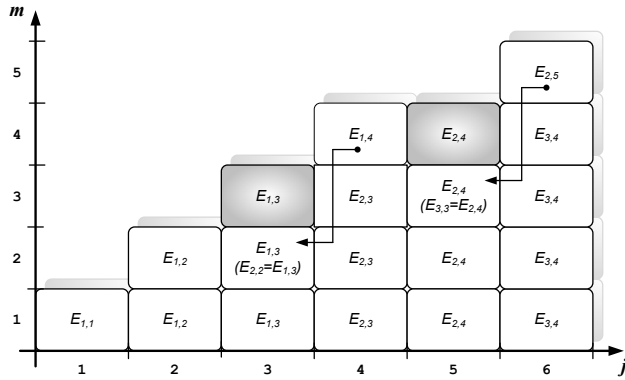
STEP 10. GOTO STEP 5.

STOP. Optimal WP tree structure for the i th input frame of the audio signal is $E_{i,m-1}$, $m = m - 2$, $l = l - 2$, $i = i + 1$. GOTO STEP 5.

The processing of the first six frames for the pipeline WP-processor is shown on

figure 7; j is a number of the frame which was loaded to WP processor for processing according to the WP tree struc-

ture $E_{m,i}$ of the current frame i . Darkened areas show where the optimal WP tree for the each frame was found, and the arrows redefine the new WP tree structure for the next current frame, which the information density $WTE_{E_{m,i}}$ of the WP tree structure is estimated on.



Initialization: $E_{0,0}, l = 0, m = 1, j = 1, i = j = 1$.

1. $l = 1, m = 2$, parent WP tree $E_{1,1}$, child WP tree $E_{1,2}$, $H_{E_{1,2}} < H_{E_{1,1}}$, $j = 2$
2. $l = 2, m = 3$, parent WP tree $E_{1,2}$, child WP tree $E_{1,3}$, $H_{E_{1,3}} < H_{E_{1,2}}$, $j = 3$
3. $l = 3, m = 4$, parent WP tree $E_{1,3}$, child WP tree $E_{1,4}$, $H_{E_{1,4}} > H_{E_{1,3}}$, no
 $E_{1,3}$ – optimal WP tree, $m = m - 2 = 2, i = i + 1 = 2$,
 $E_{2,2} = E_{1,3}, l = l - 2 = 1$
4. $l = 2, m = 3$, parent WP tree $E_{2,2}$, child WP node $E_{2,3}$, $H_{E_{2,3}} < H_{E_{2,2}}$, $j = 4$
5. $l = 3, m = 4$, parent WP tree $E_{2,3}$, child WP tree $E_{2,4}$, $H_{E_{2,4}} < H_{E_{2,3}}$, $j = 5$
6. $l = 4, m = 5$, parent node $E_{2,4}$, child WP node $E_{2,5}$, $H_{E_{2,5}} > H_{E_{2,4}}$, no
 $E_{2,4}$ – optimal WP tree, $m = m - 2 = 3, i = i + 1 = 3$,
 $E_{3,3} = E_{2,4}, l = l - 2 = 2$
7. $l = 3, m = 4$, parent WP tree $E_{3,3}$, child WP node $E_{3,4}$, $H_{E_{3,4}} < H_{E_{3,3}}$, $j = 6$

Figure 7 – Diagram of the dynamic reconfiguration WP tree structures and multi-frame processing in the parallel-pipeline WP processor

5. CONCLUSION

In the given paper the dynamic reconfigurable lifting based adaptive WP processor was presented. The lifting scheme allows to reduce on half the number of multiplications and summations and increase the processing speed. Applying DAT-based approach as the design techniques for time-varying WP decomposition allows us to construct dynamically adapted to input signal WP analysis.

Reconfigurable systems offers several advantages over competing alternatives: faster and smaller than general purpose hardware solutions; lower development cost than dedicated hardware solutions; dynamic reconfigurable supports multiple algorithms within a single application; multi-purpose architecture generates volume demand for a single hardware design.

The proposed techniques optimize system performance, and, in addition, provides a convenient framework within which ongoing research in the areas of non-uniform filter bank applied to speech/audio coding algorithms and reconfigurable architectures can be synergistically combined to enable the design of reconfigurable high-performance DSP systems.

6. ACKNOWLEDGMENT

This work was supported by Belarusian republican fund for fundamental research under the grant T08MC-040.

REFERENCES

1. R. Coifman, M. Wickerhauser, "Entropy based algorithms for best basis selection", *IEEE Trans. Information Theory*, vol. 38, Mar. 1992. – pp.712-718.
2. S. Mallat, "A Theory of Multiresolution signal decomposition: the wavelet representation", *IEEE transactions on pattern analysis and machine intelligence*, vol. 11 no 7, July 1989. – pp. 674-693.
3. M.V. Wickerhauser, "Adapted wavelet analysis from theory to software", A Peters Wellesley, Massachusetts, 1994. – 486p.
4. I. Cohen, S. Raz, D. Malah, "Orthonormal shift-invariant adaptive packet decomposition and representation" *Signal Processing*, 57 (3), March 1997. – pp. 251-270.
5. W. Sweldens, "The lifting scheme: A construction of second generation wavelets", *Siam J. Math. Anal.*, 29 (2) (1997). – pp. 511-546.
6. F. Arguello, J. Lopez, M. A. Trenas, E. L. Zapata, "Architecture for wavelet packet transform based on lifting steps", *Parallel Computing*, 28 (2002). – pp. 1023-1037.
7. C. Wang, W. S. Gan, "Efficient VLSI Architecture for Lifting-Based Discrete Wavelet Packet Transform", *IEEE Transactions on circuits and system - II: Express briefs*, Vol. 54, No. 5, May 2007. – pp. 422-426.
8. Al. Petrovsky, A. Petrovsky, "Dynamic algorithm transforms for reconfigurable real-time audio coding processor", *The proc. of the Intern. Conference on Parallel Computing in Electrical Engineering, PARELEC'02*, Warsaw, Poland, Sep. 22-25, 2002, - IEEE Computer Society Press, Los Alamitos, California, 2002. – pp.422-424.
9. K.K.Parhi, "Algorithm transformation techniques for concurrent processor", *Proc. IEEE*, vol. 77, Dec. 1989, - pp. 1879-1895.
10. J.G.Ackenhusen, "Real-time signal processing: design and implementation of signal processing systems", Printice Hall, NJ, 1999. - 460p.
11. J.Villasenor, B.Hutchings, "The flexibility of configurable computing", *IEEE Signal Processing Mag.*, Sep. 1998, - pp.67-84.
12. Al.Petrovsky, A.A.Petrovsky, "Audio coding with a masking threshold adapted wavelet packet based on run-time reconfigurable processor architecture", *In 110th AES Convention preprint № 5298*, 12-15 May, Amsterdam, Nederland, 2001. – 8p.
13. J.D.Johnston, "Transform coding of audio signals using perceptual noise criteria", *IEEE Trans. On Select. Areas Commun.*, vol. 6, Feb. 1988. - pp. 314-323.
14. Al. Petrovsky, "A multiresolution auditory model using adaptive WP excitation scalograms", *ELEKTRONIKA*, PAN, Warsaw, Vol. 49, №4, 2008. - p.65-70.
15. I. Daubechies, W. Sweldens, "Factoring wavelet transforms into lifting steps". *Journal of Fourier Anal. Appl.*, vol. 4, N 3, 1998. - pp. 247-269.
16. M. Coors, and etc. "Design and DSP implementation of fixed-point systems", *EURASIP journal on applied signal processing*, 2002:9. - pp. 908-925.
17. D. Menard, D. Chillet, and O. Sentieys "Floating-to-fixed point conversion for digital signal processors", *EURASIP journal on applied signal processing*, Vol. 2006, article ID 96421. - pp. 1-19.
18. X. Wu, Y. Li and H. Chen, "Programmable wavelet packet transform processor", *IEEE Electronics letters*, Vol. 35, No. 6, March 1999. - pp. 449-450.