MIMETIC WAVELET-PACKET TRANSFORM BASED ADAPTIVE ALGORITHM
FOR SPARSE RESPONSE IDENTIFICATION

Odair A. Noskoski* and José C. M. Bermudez‡

* Federal Institute for Technology Education Sul-Rio-Grandense, RS, Brazil
† Fed. Univ. of Santa Catarina, SC, Brazil
Coordinate Sciences of the Nature, Mathematics and its Technologies
odair@cefetrs.tche.br, j.bermudez@ieee.org

ABSTRACT
This paper proposes a new wavelet-packet transform based adaptive algorithm for sparse response identification. The distinctive features of the new algorithm are the on-line adaptability of the discrete wavelet packet transform (DWPT) and an efficient weight deactivation/activation schedule. The new algorithm, called mimetic wavelet-packet based (MWPB) algorithm, generalizes the WPB algorithm presented in [1]. The MWPB algorithm presents better estimation and tracking performances than existing wavelet-based sparse response identification algorithms. Monte Carlo (MC) simulation results compare the performances of the four most recently proposed algorithms in this class.

Index Terms— adaptive filters, sparse response identification, echo cancellation

1. INTRODUCTION
Identification of systems with sparse impulse responses encounters many practical applications in wireless communications, echo cancellation, underwater acoustics and geophysics. A sparse impulse responses is defined here as an impulse response that contains a large number of zero weights [2]. Several algorithms have been proposed to exploit sparsity to improve identification efficiency. A good overview of these techniques can be found in [3]. One successful approach is to locate the significant (active) samples of the unknown response [1, 2, 4–8].

The solution in [4] has the maximum number of active taps as a design parameter. [5] uses a least-squares (LS) based active tap detection scheme that may fail for impulse responses with large dynamic ranges [3]. The work in [2] was the first to exploit the wavelet transform (WT) time hierarchy in sparse system identification. The Haar-Basis (HB) algorithm in [2] works from a fixed control scale for which all weights are adapted. Weight activations are based on a fixed control scale, and the algorithm may fail for sparse responses which are not rich enough in frequency content [3]. Examples can be found among typical echo path responses [1, 7, 9]. The required adaptation of all the control weights may lead to longer than necessary adaptive filters, increasing computational cost and convergence time. [6, 8] uses two short adaptive filters. One filter operates in a partial-Haar transform domain to estimate the location of the peak of the unknown response. The second filter is centered about this estimate to perform identification in the time domain. The minimum number of adaptive coefficients is pre-determined by the chosen time-domain adaptive filter length.

WTs establish fixed subband decompositions. Wavelet packets generalize the wavelet theory and allow for flexible frequency domain signal representations [10]. Ref. [1] proposed a wavelet-packet-based (WPB) algorithm for the identification of sparse impulse responses with arbitrary frequency spectra. Frequency localization of the discrete wavelet packet transform (DWPT) is adaptively designed in an initial adaptation phase to match the spectral energy distribution of the unknown response. Time information is then used to adaptively determine the active weights. WPB algorithm was shown to match the excellent results of [2] in number of weights effectively adapted, convergence speed and steady-state mean-square deviation (MSD) whenever the algorithm of [2] provided good performance. Moreover, WPB provided very good results for responses not suitable for the WT-based approach of [2].

Phase 1 of WPB [1] is dedicated to the one-time design of the DWPT based on rough adaptive estimations of the unknown response. Phase 2 is dedicated to the detection and estimation of the significant weights. Phase 1 is the most critical of WPB, as it requires a minimum number of iterations using all adaptive weights to define the DWPT. A modification to WPB has been recently proposed [7] which uses a region-based strategy for adaptive weight activation and a new deactivation/activation schedule across the transform scales. The resulting region-based wavelet-packet algorithm (RBWP) significantly reduces the time required to build the wavelet-packet transform and improves robustness to design parameters when compared to previous solutions.

This paper builds on the good properties of RBWP and proposes a new algorithm called Mimetic Wavelet-Packet Based (MWPB) that improves RBWP in two main aspects: 1) it allows the modification of the DWPT on the fly to better match the properties of the unknown response, including nonstationarities; 2) a new weight deactivation/activation (D/A) schedule allows for weight estimate refinement over several adaptation cycles, which improves overall performance. Simulation results show improved performance of MWPB, as compared to HB, WPB and RBWP [1, 2, 7]. The algorithm has been successfully tested with all echo responses in [9] and with some synthetic responses.

This work was partially supported by CNPq under grant No. 308095/2003-0 and by Capes.
2. ADAPTATION STRATEGY

One important aspect of building the DWPT on the fly is initialization. MWPB can be initialized using different strategies. One possibility is to start with an arbitrary complete DWPT structure and adapt all the coefficients. Another strategy (used in this paper) is to start adapting only few levels of an arbitrary initial DWPT structure, progressively completing the other DWPT levels as identification continues. The definition of the initial DWPT structure can benefit from available information about the unknown system. For instance, it has been found that a good initialization for network echo cancellation is to subdivide the frequency bands progressively towards the center of the frequency spectrum. This was the initialization used in the example presented in this paper. Also, MWPB can be used with any adaptive algorithm. We here use NLMS because of its popularity and to facilitate comparisons with [1, 2, 7].

The MWPB algorithm is schematically described in Fig. 1. After initialization with an arbitrary DWPT, the algorithm starts an adaptation interval (AI). Each AI is followed by a weight D/A step (Section 3). An adaptation cycle is defined as a set of $M$ AIs, where $M$ is the number of levels in the DWPT. After a number of adaptation cycles, the DWPT structure is evaluated and a decision is made as to whether it should be adjusted or maintained. These steps are repeated indefinitely or until a stopping criterion defined by the designer is met.

Fig. 1. Schematic description of the MWPB algorithm. $N_c$ is the number of adaptation cycles after which the DWPT structure is evaluated.

Fig. 2 shows the block diagram of the MWPB adaptation structure. $N$ is the number of weights effectively adapted at each AI. The $N$-tap input vector is $x(n) = [x(n), \ldots, x(n-N+1)]^T$. Thus, the DWPT transformation matrix is $\tilde{N} \times N$. The $\tilde{N} \times 1$ vector of transformed signal samples is $\hat{z}_a(n)$, and this vector excites the adaptive filter with weight vector $\tilde{w}_a(n)$. The adaptive weight vector $\tilde{w}_a(n)$ contains the active weights of all DWPT levels $m = 1, \ldots, M$. $y(n)$ is the desired signal, $\hat{y}(n)$ its estimate and $e(n)$ is the estimation error. Vector $\tilde{z}_a(n)$ can be written as $\tilde{z}_a(n) = [\hat{z}_{a_1}^T(n), \hat{z}_{a_2}^T(n), \ldots, \hat{z}_{a_M}^T(n), \hat{z}_{a_{M+1}}^T(n)]^T$, where $\hat{z}_{a_m}(n)$ is the transformed signal vector at the $i$-th DWPT level (note that the last level has 2 vectors $\hat{z}_{a_{M+1}}(n)$ and $\hat{z}_{a_{M+2}}(n)$).

\[ \tilde{w}_a(n+1) = \tilde{w}_a(n) + 2\tilde{\lambda}_a^{-1}(n)e(n)\hat{z}_a(n) \]  

where $e(n) = y(n) - \hat{y}(n)$ and $\hat{y}(n) = \hat{z}_a^T(n)\tilde{w}_a(n)$. $\tilde{\lambda}_a^{-1}(n)$ is a diagonal matrix whose elements $\tilde{\lambda}_{a_m}^{-1}(n)$ are estimates of the transformed input signal power. These estimates are determined through the exponential averaging [2]

\[ \tilde{\lambda}_{a_m}^{-1}(n) = (1-\alpha)\tilde{\lambda}_{a_m}^{-1}(n-1) + \alpha \tilde{z}_{a_m}^2(n), \quad i = 1, \ldots, M+1 \]  

where $i = 1, \ldots, M-1$ is the DWPT level index ($i = m$) for the present AI. Indexes $i = M$ and $i = M+1$ both refer to level $m = M$. $0 < \alpha < 1$ is the smoothing factor that depends on the stationarity of the input signal, and $\tilde{z}_{a_m}(n)$ is the first significant element of the $i$-th transformed input vector [2]. The step-size $\tilde{\mu}$ is inversely proportional to the effective number of weights $N \tilde{\lambda}_a(n)$.

3. WEIGHT DEACTIVATION/ACTIVATION STRATEGY

After each AI, MWPB performs a weight D/A step. The adaptive weights are compared to the threshold $TH = \beta_{\alpha} \sqrt{\tilde{\mu} \tilde{\xi}(k)/\tilde{\lambda}_a^2(k)}$ [2]. Here $\tilde{\xi}(k) = (1-\rho)\tilde{\xi}(k-1) + \rho e^2(k)$ is an instantaneous estimate of the mean-square error $E[e^2(k)]$, $\tilde{\mu}/\tilde{\lambda}_a^2(k)$ is the normalized step size for NLMS and $\beta_{\alpha}$ is a parameter that determines the probability of false alarm [2]. $\rho = \tilde{\mu}$ is assumed to facilitate comparisons with [2].

Fig. 3 illustrates the temporal hierarchy of a 5-level Haar-based DWPT ($M = 5$). All dark rectangles represent transform coefficients in the same temporal hierarchy of element (3,2) (marked with *). The horizontal direction shows the coefficient distribution in time. The vertical direction shows the 5 transform levels ($m = 1, \ldots, 5$). Each rectangle corresponds to the region of greater influence of a transform coefficient.

Each AI is followed by a weight D/A step. These operations can follow different schedules. Table 1 describes three possibilities. One adaptation cycle is comprised of one AI for each transform level. In our example there are 5 AIs in each cycle. The numbers in the table refer to the level number $m$ in which weights are tested for D/A.

Table 1. Three possible deactivation/activation schedules after each AI in one adaptation cycle for $M = 5$.

<table>
<thead>
<tr>
<th>AI in each cycle</th>
<th>AI1</th>
<th>AI2</th>
<th>AI3</th>
<th>AI4</th>
<th>AI5</th>
</tr>
</thead>
<tbody>
<tr>
<td>I Deact. in level</td>
<td>1</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Act. in level</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>II Deact. in level</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>Act. in level</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>III Deact. in level</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>Act. in level</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>2</td>
</tr>
</tbody>
</table>

Fig. 3. Hierarchic structure of the DWPT-Haar.
Consider, for instance, the weights corresponding to transform level 2 (m = 2). In Schedule I, weights in this level can be activated after AI1 and are tested for deactivation after AI2, after only one adaptation interval. Thus, each AI should be long enough for an accurate weight estimation. Otherwise, the quality of the decision in the deactivation may be low. On the other hand, the D/A decisions are quickly propagated through the transform levels. WPB uses Schedule I. In HB [2] all control weights are tested and D/A is performed at all levels after each AI (except the control level). In Schedule II of Table 1, activation in level 2 is done after AI1 and these weights are tested for deactivation after AI2 of the next cycle (after 4 AIs). This is the opposite extreme, compared to Schedule I. The AIs can be the shortest, as each weight is allowed 4 AIs to converge. However, the decisions propagate slowly through the transform. RBWP uses Schedule II. Schedule III (used in this work) is a compromise between these two extremes.

4. ADJUSTMENT OF THE DWPT STRUCTURE

MWPB allows periodic adjustment of the DWPT. As done in [1, 7], only either the lower or the upper subband is allowed to be subdivided at each transform level. This structural restriction allows the use of the temporal hierarchy and facilitates real-time processing. The filters used are Haar filters $H_L$ (lowpass) and $H_H$ (highpass). This section describes the proposed adjustment of the DWPT structure.

The DWPT is initialized with any structure satisfying the structural restrictions just described. After some, say $N_c$, cycles of AIs, the DWPT structure is evaluated for redesign given the present estimate of unknown impulse response. To this end, the estimated response vector $\hat{w}_a(n)$ is first determined by inverse DWPT transforming $\hat{w}_a(n)$. Then, a new DWPT is designed for the input $\hat{w}_a(n)$, as shown hypothetically in Fig 4 for a 3-level transform. This design is not adaptive. It simply determines the restricted DWPT structure that fits best the frequency spectrum of $\hat{w}_a(n)$. As in [1, 7], only the subband with larger energy is subdivided at each DWPT level. The decimated filter outputs at each level compose the new weight vector $\hat{w}_a(n)$. In the example in Fig 4, $\hat{w}_a(n) = [\hat{w}_a^H(n), \hat{w}_a^L(n), \hat{w}_a^{H2}(n), \hat{w}_a^{L2}(n)]$. After definition of the new DWPT, a detection step determines the new vector $\hat{w}_a(n)$ of significant weights. Next, MWPB resumes the adaptation process using the new DWPT structure.

5. COMPUTATIONAL COMPLEXITY

Analytical determination of the exact computational complexity of HB, WPB, RBWP, or MWPB is very difficult, as the number of adapted weights varies according to decisions made during the identification process. Compared to HB, algorithms WPB, RBWP and MWPB have the extra complexity required for building the DWPT. Complexities required for detections, decisions and step-size updating are implementation dependent and their cost is typically smaller than the cost of a single iteration. Finally, the cost of building the DWPT using the MWPB algorithm is less than the cost of two such iterations. In comparing HB with WPB, RBWP and MWPB, note that HB performs a complete weight detection and D/A after each AI. WPB, RBWP and MWPB perform these steps after each AI only for the weights of two levels (m = 2). MWPB algorithm has the additional cost of the periodic DWPT re-evaluation. Table 2 shows the number of operations required for a typical iteration of the HB, WPB, RBWP and MWPB algorithms as a function of $N$ and $N_c$. Well known results for NLMS are also shown as a reference [11]. The value of $\tilde{N}$ ($\tilde{N} \geq N$) will in general be different for each algorithm (see Section 6 for examples).

6. SIMULATION

This section compares the performances of the HB, WPB, RBWP, MWPB and MWPB(1), where MWPB(1) is MWPB using the region-based weight detection proposed for RBWP [7] with a single weight aggregated to each side of each detected weight. Identification starts with a 512-tap sparse impulse response $h_1(n)$ (model $g_{in}$ from [9]). At iteration 60,000, $h_1(n)$ is replaced by a synthetic response $h_2(n)$, which was designed to move the energy concentration from the low band in $h_1(n)$ to a high band to test the tracking ability of the algorithms. Fig. 5 shows the effective parts of $h_1(n)$ and $h_2(n)$ and the frequency spectra of these responses.

The input sequence $x(n)$, with variance $\sigma_x^2 = 1$, was generated using a filter with transfer function $H(z) = 0.25\sqrt{3}/(1-1.5z^{-1}+z^{-2}-0.25z^{-3})$ excited by a white Gaussian sequence [2, Eq. (29)]. A white Gaussian measurement noise with variance $\sigma^2 = 10^{-4}$ was added to $y(n)$. Step sizes used were $\mu = 1/10N$ for WPB, RBWP, MWPB and MWPB(1) and $\mu = 1/N$ for HB. The different step sizes are due to the reduced values of $\tilde{N}$, as compared to HB. AIs with $4\tilde{N}$ iterations were used for all algorithms, except in the Phase 1 of WPB and in AI1 of RBWP. For Phase 1 in WPB, AI1 and AI2 had 2000 and 4000 iterations, respectively. AI1 in Phase 1 of RBWP had 200 iterations. DWPT structure re-evaluation in MWPB and MWPB(1) was done every three adaptation cycles plus one AI. The detection threshold parameter $\beta_{in}$ was 2.57 for HB, RBWP and MWPB(1) and 0.9 for both WPB and MWPB.

For WPB and MWPB(1), only the corresponding weights to levels 4–9 of the DWPT were initially adapted (64 weights out of...
This paper has proposed the mimetic wavelet-packet transform based (MWPB) algorithm for sparse response identification with arbitrary spectra. The new algorithm generalizes the adaptation strategy of previous algorithms. The new strategy allows the selection of the discrete wavelet packet transform that best matches the unknown spectrum under certain practical restrictions. Simulation results have shown faster convergence, better steady-state performance and greater robustness to the changes in unknown response when compared to existing algorithms.

7. CONCLUSION

Fig. 5. (a) Response $h_1(n)$ (b) Frequency spectrum of $h_1(n)$ (c) Response $h_2(n)$ (d) Frequency spectrum of $h_1(n)$.

Fig. 6. Number of weights effectively adapted (NWEA).

Fig. 7. Excess MSE (EMSE) evolution.

512). Then, AIs $A_{I_1}$, $A_{I_2}$ and $A_{I_3}$ of schedule I in Table 1, with the corresponding weight detections, deactivations and activations were performed. After this, all levels had their active weights detected for the first time. This concluded initialization. From this point on the algorithms moved into regular operation and followed D/A schedule III in Table 1.

slower in recovering their EMSE performances. This happens basically because WPB uses the D/A schedule I in Table 1 and both WPB and RBWP use a fixed DWPT structure determined once in Phase 1. Moreover, the algorithms MWPB and MWPB(1) achieved excellent results before and after the change in unknown response. This was due to the on-line adaptability of the DWPT and the use of the D/A schedule III in Table 1.

8. REFERENCES