LOW-COMPLEXITY ROBUST SPARSE CHANNEL IDENTIFICATION USING PARTIAL BLOCK WAVELET TRANSFORMS-ANALYSIS AND IMPLEMENTATION

*Celso H. H. Ribas**, *José C. M. Bermudez*‡ *and Neil J. Bershad*†

* Acústica Amplivox, Florianópolis, SC, Brazil ‡ Fed. Univ. of Santa Catarina, Dept. of Elec. Eng., Florianópolis, SC, Brazil
† Univ. of California Irvine, Dept. of Elec. Eng. and Comp. Science, Irvine, CA, USA
celsoribas@amplivox.com.br, j.bermudez@ieee.org, bershad@ece.uci.edu

ABSTRACT

This paper presents a novel implementation for identifying sparse telephone network echo channels. The new scheme follows the approach used in [1] in that the location of the channel response peak is estimated in the wavelet domain. A short time-domain adaptive filter is then located about the estimated peak to identify the sparse response. The primary purpose of this paper is to present an efficient design of such system. The use of a new block wavelet transform results in both 70% less computational complexity and improved peak detection. A new robust time-domain adaptive filtering is also proposed which significantly reduces the jitter problem in [1]. Monte Carlo simulations show excellent echo cancellation for a typical ITU-T channel.

***Index Terms*—** echo cancellation, adaptive filtering

1. INTRODUCTION

Sparse impulse responses are encountered in many applications [3]. Network echo cancellation is one important example. The channel bulk delay is often much longer than the echo path impulse response (the nonzero portion of the channel impulse response). Typical bulk delays can be on the order of 128 ms [1]. Most dispersion times of network echo path impulse responses (duration of the nonzero response) are between 5 and 7 ms [2]. Thus, it is impractical to use full length adaptive filters to identify the channel impulse response since long adaptive filters are both slow to adapt and have noisy weights [4]. Several structures and algorithms have been proposed for efficient network echo cancellers which exploit the channel impulse response sparseness [1, and references therein].

The solution in [1] uses two short adaptive filters operating sequentially. The first adaptive filter adapts in the wavelet domain using a partial Haar transform of the input at each time sample. This filter yields an estimate of the location of the peak of the channel impulse response. A second short time-domain adaptive filter is then centered about this bulk delay estimate to indentify the short echo path impulse response. Both filters are adapted using the LMS algorithm [1] but other algorithms may be used. Thus, two short adaptive filters can be used instead of a very long one, resulting in fast overall convergence and reduced computational complexity and storage.

The solution in [1] represents an effective way to identify sparse network echo responses. However, its performance is influenced by important design issues: 1) computational complexity for implementing the wavelet transform; 2) centering of the time-domain adaptive filter about the estimated peak; and 3) as the peak estimate changes, tracking of the bulk delay leads to a jitter problem. These three design issues were not studied in [1]. Note that, although the basic system studied here is the same as in [1], the implementation is quite different (and novel) and leads to significant performance improvement.

This work was partially supported by CNPq under grant No. 308095/2003-0 and scholarship No. 131923/2004-8.

This paper presents a novel scheme for identifying sparse impulse responses which is based on [1]. However, the sparse response bulk delay is estimated in the wavelet domain using a Daubechies spline wavelet transform on both the input and the desired signals (as compared to the Haar transform only on the input signal in [1]). Block processing of the transforms yields computational savings of more than 80% (compared to [1]) for the peak detection. The peak detection is also improved in the wavelet domain because the new wavelet transform displays more clearly the impulse response peak. Finally, a more robust approach is presented for the time-domain adaptive filter window location and update. This new approach significantly reduces the jitter problem [1]. When compared to [1], the new scheme leads to significant performance improvement and to computational savings of more than 50% for the complete solution. A Monte Carlo simulation example shows excellent results for echo cancellation for a typical impulse response (ITU-T recommendation G.168 [2]).

2. THE PARTIAL HAAR TRANSFORM

The Haar discrete wavelet transformation of a vector of length 2^r can be represented by its pre-multiplication by a $2^r \times 2^r$ matrix $\boldsymbol{H}$ [5]. The partial Haar transformation in [1] uses only a single scale of the wavelet, say the m-th. The corresponding matrix $\boldsymbol{H}_m$ is a $2^{r-m} \times 2^r$ submatrix of $\boldsymbol{H}$ whose elements are defined as

$$\boldsymbol{H}_m(i,j) = \psi_m\left[j-(i-1)2^m-1\right], \tag{1}$$

where

$$\psi_m(\ell) = \begin{cases} 2^{-\frac{m}{2}}, & 0 \leq \ell \leq 2^{m-1}-1 \\ -2^{-\frac{m}{2}}, & 2^{m-1} \leq \ell \leq 2^m - 1 \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

As an example, for $r = 3$ and $m = 2$, the partial Haar matrix is given by

$$\boldsymbol{H}_2 = \begin{bmatrix} 0.5 & 0.5 & -0.5 & -0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0.5 & -0.5 & -0.5 \end{bmatrix} \tag{3}$$

The choice of m determines the number of coefficients of the transform domain adaptive filter. The transform domain adaptive filter uses only 2^{r-m} coefficients for an unknown response with length 2^r. This adaptive filter cannot model exactly the 2^r-long impulse response. Thus, the following scheme estimates the approximate location of the channel impulse response peak. Note that every row of $\boldsymbol{H}_m$ is composed of the same nonzero base vector ${\boldsymbol{h}_m}^T$ and additional zeros. In (3), $\boldsymbol{h}_2^T = [0.5, 0.5, -0.5, -0.5]$. This property can be used to reduce complexity. In general,

$$ {\boldsymbol{h}_m}^T = [\psi_m(0), \ldots, \psi_m(2^m - 1)], \quad (4) $$

which is the wavelet in the m-th partial.

3. THE PARTIAL BLOCK DWT LMS ALGORITHM

3.1. The New Scheme

The computational complexity of [1] is mainly due to computing the partial Haar transformed input vector at each time instant. This is because the transformed vector does not have a shift structure[1]. Thus, the entire transformed vector must be evaluated at each iteration. Only the most recent transformed output sample would need to be evaluated at each iteration if the transformed vector had a shift structure. This property holds if the time domain signals are processed in blocks of length 2^m. For example in eq. (3), the input vector of length 8 is composed of two subvectors of length 4. Each subvector is multiplied by the same base vector ${\boldsymbol{h}_m}^T$ to generate the two samples of the transformed vector. If the input vector samples advance in blocks of length 4, only one new transform sample must be evaluated at each iteration. Thus, the transform vector will then have the shift structure.

Fig. 1 shows the block diagram of the proposed implementation. $x(n)$ is the input signal, $\boldsymbol{w}_o$ is the unknow parameter vector, $\eta(n)$ is the additive noise, $d(n)$ is the desired signal, $\boldsymbol{w}(k)$ is the wavelet-domain adaptive weight vector, $\boldsymbol{w}_t(n)$ is the time-domain adaptive weight vector, n is the time sample index and k is the block sample index. The input vector to the wavelet-domain adaptive filter $\boldsymbol{z}(k)$ has the shift structure. The top element of $\boldsymbol{z}(k)$ and the vector $\boldsymbol{w}(k)$ are updated only at each block of 2^m samples of $x(n)$, resulting in a block adaptive filtering structure. Thus, the estimated peak location remains invariant during the entire block. This estimate determines the location of the time-domain adaptive filter $\boldsymbol{w}_t(n)$.

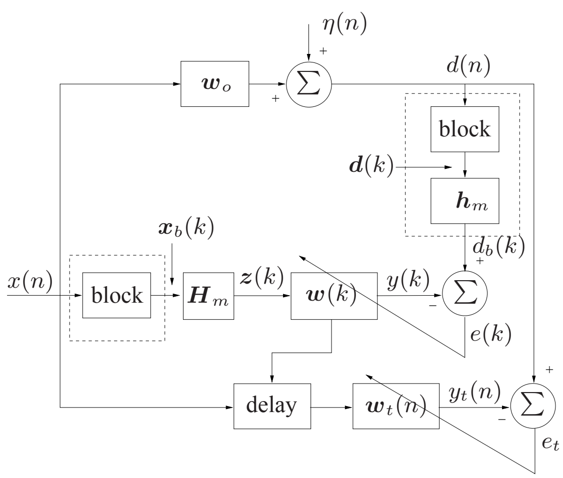

Fig. 1. Scheme of the proposed implementation.

[1]A shift structure is observed in vectors that arise from a tap-delay line implementation.

3.2. The Block Wiener Solution

From Figure 1,

$$ \boldsymbol{z}(k) = \boldsymbol{H}_m \boldsymbol{x}_b(k) \qquad k = 0, 1, 2... \quad (5) $$

where k refers to the block index,

$$ \boldsymbol{x}_b(k) = \big[x[(k+1)2^m - 1], \ldots, x[(k+1)2^m - 2^r]\big]^T \quad (6) $$

and

$$ \boldsymbol{z}(k) = \big[z(k), z(k-1), \ldots, z(k - 2^{r-m} + 1)\big]^T . \quad (7) $$

with $x(n) = 0$ for $n < 0$. The desired signal $d(n)$ is also processed in blocks. The desired signal block vector $\boldsymbol{d}(k)$ is

$$ \boldsymbol{d}(k) = \Big[d[(k+1)2^m - 1], \ldots, d[(k+1)2^m - 2^m]\Big]^T . \quad (8) $$

where

$$ d(n) = \eta(n) + \boldsymbol{x}^T(n)\boldsymbol{w}_o, \quad (9) $$

and $\boldsymbol{x}(n) = [x(n), x(n-1), \ldots, x(n - 2^r + 1)]^T$.

The transform $d_b(k)$ of $\boldsymbol{d}(k)$ is then obtained as

$$ d_b(k) = {\boldsymbol{h}_m}^T \boldsymbol{d}(k). \quad (10) $$

Using the orthogonality principle, $E\{\boldsymbol{z}(k)e(k)\} = \boldsymbol{0}$ and the transform domain Wiener filter is given by

$$ \boldsymbol{w}_{o_P} = E\{\boldsymbol{z}(k)\boldsymbol{z}^T(k)\}^{-1} E\{\boldsymbol{z}(k)d_b\}. \quad (11) $$

Assuming that $x(n)$ and $\eta(n)$ are statistically independent i.i.d. zero-mean Gaussian sequences and noting that $\boldsymbol{x}_b(k) = \boldsymbol{x}\big[(k+1)2^m - 1\big]$, it can be shown that

$$ \boldsymbol{w}_{o_P} = \boldsymbol{H}_m \boldsymbol{L}_m \boldsymbol{w}_o = \boldsymbol{S}_m \boldsymbol{w}_o, \quad (12) $$

where $\boldsymbol{S}_m = \boldsymbol{H}_m \boldsymbol{L}_m$ and $\boldsymbol{L}_m$ is a $2^r \times 2^r$ matrix defined as

$$ \boldsymbol{L}_m(i,j) = \begin{cases} \psi_m(i-j), & j \le i \le j + 2^m - 1 \\ 0, & \text{otherwise.} \end{cases} \quad (13) $$

The Wiener solution was given in [1] by $\boldsymbol{H}_m \boldsymbol{w}_o$ for the same input signal and noise. Alternatively, Eq. (12) shows that the block wavelet-domain adaptive filter operates in the $\boldsymbol{S}_m$ domain, not in the partial-Haar ($\boldsymbol{H}_m$) domain. The basis vector $\boldsymbol{s}_m$ for matrix $\boldsymbol{S}_m$ is the autocorrelation function of the basis vector $\boldsymbol{h}_m$ associated with $\boldsymbol{H}_m$.[2] It can be verified that $\boldsymbol{S}_m$ corresponds to the Daubechies' bi-orthogonal 2.2 spline wavelet [6] without energy normalization. The partial block transform $\boldsymbol{S}_m$ has one important advantage over the partial Haar transform $\boldsymbol{H}_m$. $\boldsymbol{S}_m$ has better properties for estimating the location of the peak of the unknown response $\boldsymbol{w}_o$. Fig. 2 shows the "continuous" version of $\boldsymbol{h}_m$ and $\boldsymbol{s}_m$ for $m = 3$. $\boldsymbol{s}_m$ is similar to the typical peaked channel impulse responses in network echo cancellation whereas $\boldsymbol{h}_m$ is not [2]. Thus, $\boldsymbol{S}_m$ yields larger absolute values of the transform about the peak of $\boldsymbol{w}_o$, and thus improves detection of the peak.

[2]Considering the sequence $\psi_m(0), \ldots, \psi_m(2^m - 1)$, the $2^{m+1} - 1$ elements from $\boldsymbol{s}_m$ will be the nonzero samples of the autocorrelation function $r_{\boldsymbol{s}_m}(l) = \sum_{\kappa=-\infty}^{+\infty} \psi_m(\kappa)\psi_m(\kappa - l), \; -2^m \le l \le 2^m$.

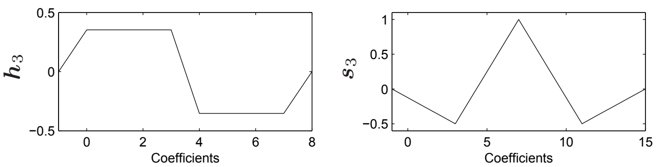

Fig. 2. Base vectors of $\boldsymbol{H}_m$ and $\boldsymbol{S}_m$.

3.3. Location of the Time Domain Adaptive Filter

This section describes a new approach for locating the short time-domain adaptive filter. The approach uses the transform-domain adaptive filter estimates. Typical responses $\boldsymbol{w}_o$ have fast rising and exponentially decaying peaks. Thus, pre-multiplication by $\boldsymbol{S}_m$ spreads and right-shifts the channel impulse response $\boldsymbol{w}_o$ peak in the transform domain. Fig. 3 illustrates this property for the echo path model $g_5(n)$ given in [2].

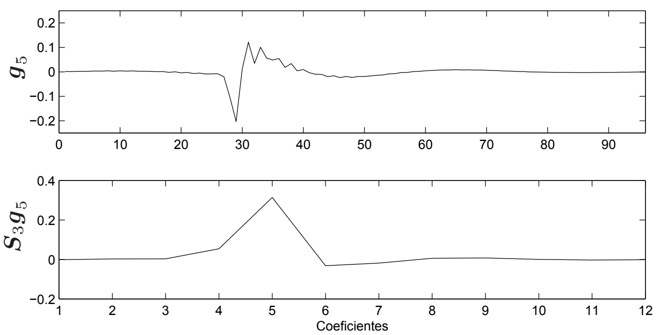

Fig. 3. Model g_5 in [2] and the transformed response.

Typical echo paths [2] have peaks located at beginning of their impulse responses. The peak usually occurs before 4.25 ms (tap 34 for an 8 KHz sampling rate). Dispersion times are typically limited to 12 ms (96 taps). A good approach has been determined experimentally to align the peak about the $(34+2^m+2^{m-1})$-th coefficient of the 128-tap time domain adaptive filter[3]. The peak location must be translated to the time domain. Analysis of $\boldsymbol{S}_m$ shows that the coefficient of $\boldsymbol{w}_{o_P}$ at position c maps into a region centered about the position $(c-1)2^m+1$ in the time domain response $\boldsymbol{w}_o$. This approach compares favorably with the approach proposed in [1]. The latter approach centered the time-domain filter about the estimated peak and neglected that the peak usually occurs at the beginning of the response [2].

3.4. The Jitter Problem

Effective sparse impulse response identification requires resetting the short time-domain adaptive filter weights whenever the estimated peak location changes. The adaptive filter weights were not changed in [1] after the peak estimate changed. This approach and the filter location scheme discussed in Section 3.3 led to a jitter problem. The solution in [1] was to shift the time-domain filter only if the present and the previous peak location estimates differed by more than q taps. The block implementation reduces the fluctuations in the peak location estimates. Moreover, the proposed time-domain response location approach includes all the significant echo response coefficients in the time-domain adaptive filter window. However, the initial conditions on the time domain adaptive filter should be re-synchronized after each new change in the peak location estimate. Suppose the new peak estimate changes p coefficients to right. Then, the initial p coefficients of the time-domain filter are discarded and p new coefficients are grouped at the end of the filter's response. The reverse process is performed if the change is to the left. The initial conditions of the $2^{r-m}-p$ coefficients must then be changed to the values they had at the last iteration to avoid re-initialization. The new p coefficients are initialized at zero since the initial and final samples of the echo response are usually close to zero. Time-domain adaptation can even occur during the transform-domain peak detection transient. This also improves the overall convergence rate. Note that the block delay affects only the peak estimation adaptive filter. The time-domain adaptive filter is adapted continuously. Consider, for instance, the typical case of $m=3$ and an 8kHz sampling rate. The delay in this case would be only $2^3=8$ samples. This corresponds to a 1ms delay. This delay is negligible at 8,000 samples per second.

[3] The additional delay 2^m+2^{m-1} has been experimentally determined to compensate for the right shift due to the exponential tail of $\boldsymbol{w}_o$.

4. COMPUTATIONAL COMPLEXITY

This section shows that the block implementation is computationally much more efficient than the scheme of [1]. The following is assumed: (1) the partial Haar transform in [1] is computed recursively for $m \geq 3$ (more efficient); (2) the block transform implementation is performed only every 2^m time samples. Thus, the average computational complexity per time sample is considered; and (3) the adaptive filters are updated using the LMS algorithm.

Only the first element $z(k)$ of vector $\boldsymbol{z}(k)$ in (7) must be computed at each transform evaluation in the block implementation. This is because $\boldsymbol{z}(k)$ has the shift structure. $z(k)$ can be evaluated by additions and subtractions of the samples of $x(n)$. A multiplication by $2^{-\frac{m}{2}}$ normalizes the coefficient, according to (2). The same procedure is applied for the transform of $\boldsymbol{d}(k)$.

The number of operations per input sample interval are compared in tables 1 and 2 for $\boldsymbol{w}_o$ with $N=1024$ coefficients and $m=2,3$ and 4. Table 1 shows the total number of operations (sums, subtractions and multiplications) necessary to estimate the peak location. Table 2 shows the total number of operations required for the complete solution (transform and time domains). Both tables also show the percentage of reduction in the number of operations, compared to [1].

Table 1. Operations per time sample to estimate the peak location.

m	[1]	Block Solution	Reduction
2	2049	258.25	87.4%
3	1281	66.125	94.8%
4	641	18.0625	97.2%

Table 2. Operations per time sample for the complete solution.

m	[1]	Block Solution	Reduction
2	2562	771.25	69.9%
3	1794	579.125	67.7%
4	1154	531.0625	54.0%

Note also that the computational complexity of [1] may increase significantly if a wavelet other than Haar is used. Most wavelets do not allow for recursive computations of the partial discrete wavelet transforms[4] The computational complexity of the proposed scheme would not increase significantly for different wavelets if the block lengths are defined properly.

5. MONTE CARLO SIMULATIONS

Monte Carlo simulations were run for the eight echo impulse responses given in [2] for 50,000 input samples and averaged over 50 runs. The parameters used were $\sigma_x^2 = 1$, $\sigma_\eta^2 = 10^{-6}$, and $m = 2, 3$ and 4. The step-size for the LMS algorithm in the transform domain in [1] was $\frac{0.1}{(2^{r-m}+2)\sigma_x^2}$. The step-size for the block implementation was chosen to provide the same transformed domain steady-state mean-square deviation for both implementations. The step-size for the time domain LMS adaptive filter was $\frac{0.1}{(130)\sigma_x^2}$ in both cases.

Due to space limitations, the only results shown are for the echo path impulse response $\boldsymbol{g}_2$ [2]. These results are representative of all cases studied.

Fig. 4 shows the sparse channel impulse response $\boldsymbol{w}_{o2}$ formed with $\boldsymbol{g}_2$. Fig. 5 shows the estimated peak location over time. Note that Fig. 5 does not imply that the overall new scheme is slower than the scheme in [1] for practical purposes. This is because the number of extra iterations required for peak detection is not significant compared to the typical convergence times as shown in Fig. 6. Fig. 6 presents the mean-square error (MSE) over time for the two time domain adaptive filters. This figure displays the same time domain convergence rates for each implementation even though the convergence rate is slower for the transform domain filter in the block implementation. The steady-state MSE of the proposed implementation shows a significant 20dB improvement over [1]. These results are due to the block implementation time response location and to the handling of initial conditions. Figs 7 and 8 show the estimated peak location and the MSE plotted as a function of the number of operations. These figures show that the transform domain block implementation yields important computational savings, compared to [1].

6. RESULTS AND CONCLUSIONS

This paper has presented a new block implementation scheme for identifying sparse impulse responses using the basic structure proposed in [1]. The block-based solution suggests new schemes to 1) estimate the location of the impulse response peak using a block transform adaptive filter, 2) locate the time domain adaptive filter about the estimated peak and 3) adapt the time domain filter. The new transform domain Wiener solution is more effective for estimating the peak location than the solution in [1]. The new time-domain adaptive process reduces the effects of the estimation jitter. The solution studied here yields significant reductions in computational complexity and significantly better cancellation levels for the same convergence rate of the solution in [1].

7. REFERENCES

[1] N. J. Bershad and A. Bist, "Fast coupled adaptation for sparse impulse responses using a partial haar transform," *IEEE Transactions on Signal Processing*, vol. 53, no. 3, pp. 966–976, March 2005.

[4]There is no repetition of coefficient values within the basis vectors for most wavelets.

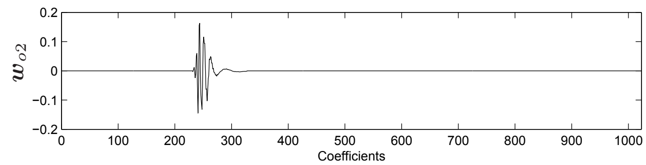

Fig. 4. Impulse response $\boldsymbol{w}_{o2}$.

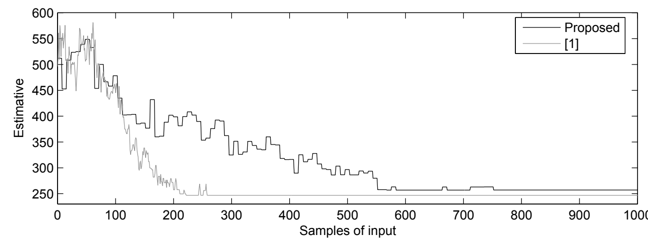

Fig. 5. Estimated location of the peak.

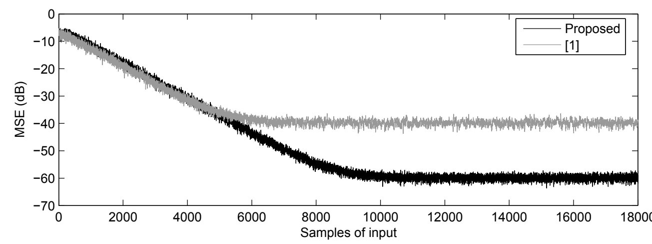

Fig. 6. MSE of the time domain adaptive filter.

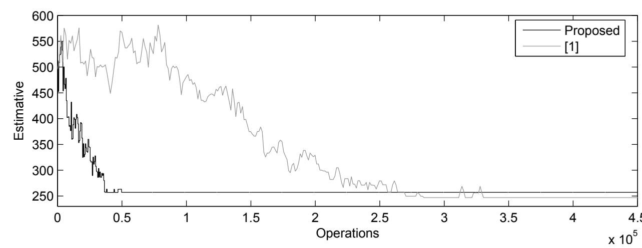

Fig. 7. Estimated location of the peak by operations.

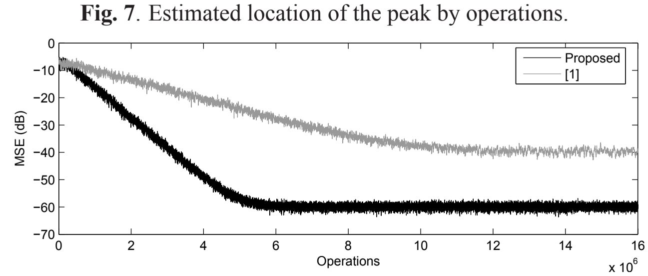

Fig. 8. MSE of the time domain adaptive filter by operations.

[2] ITU-T, *Digital Network Echo Cancellers*, Recommendation G.168, 2004.

[3] J. Benesty and Y. Huang, Eds., *Adaptive Signal Processing: Applications to Real World Problems*, Springer, Berlin, Germany, 2003.

[4] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, New Jersey, 1991.

[5] I. Daubechies, "Orthogonal bases of compactly supported wavelets," *Commun. Pure Applied Math.*, vol. XLI, pp. 909–996, 1988.

[6] I. Daubechies, *Ten Lectures on Wavelets*, SIAM, Philadelphia, Pennsylvania, 1st edition, 1992.