Sankhyā : The Indian Journal of Statistics
Special issue on Wavelets
2001, Volume 63, Series A, Pt. 3, pp. 367–393

MULTIWAVELETS AND SIGNAL DENOISING*

By SAM EFROMOVICH
University of New Mexico, Albuquerque

SUMMARY. Despite their better approximation and data compression properties, multiwavelets are less known to practitioners than uniwavelets and they are rarely used in statistical applications. The reason is that calculation of multiwavelet empirical coefficients is essentially more complicated than the familiar orthogonal uniwavelet discrete transform, in particular a special prefiltering of observations is required. As a result, even for the case of a signal contaminated by white Gaussian noise, empirical multiwavelet coefficients are contaminated by a nonstationary correlated noise. On the top of this, different types of prefilters and multiwavelets lead to different noise distributions. Thus to make multiwavelets convenient for statistical applications, denoising should be robust and self–learning. Also, statistical applications should be found where multiwavelets justify the additional efforts involved in their use. In this article a new adaptive procedure of denoising and recovering derivatives based on a vaguelette–vaguelette approach is developed. Asymptotic optimality of the estimator is established, and results of numerical simulations are presented that justify the use of the multiwavelet estimator.

1. **Introduction**

Multiwavelets have excellent approximation and data compression properties but they are practically unknown to statisticians and rarely used in statistical applications. There are two main reasons for this. The first one is that calculation of multiwavelet empirical coefficients is essentially more

Paper received July 2000; accepted July 2001.
AMS (1991) *subject classification.* Primary 62G05; secondary 62G07.
Key words and phrases. Adaptation, asymptotics, colored noise, cristina biwavelets, derivative estimation, ill-posed problem, learning, oracle inequality, regression, robustness, small datasets, vaguelette.
*This article is based on the research funded by NSF Grant 9971051.

complicated than the familiar orthogonal uniwavelet discrete transform. In particular a special prefiltering of observations is required. As a result, even for the case of a signal contaminated by white Gaussian noise, empirical multiwavelet coefficients are contaminated by a nonstationary correlated noise. Moreover, different types of prefilters and multiwavelets imply different noise distributions. The second reason is that for the classical case of Gaussian noise empirical results have not been able to demonstrate that multiwavelet estimators always outperform uniwavelet estimators.

Thus, should a statistician even consider using multiwavelets? Can multiwavelets be attractive for solving classical statistical problems? Is the additional effort associated with multiwavelets worthwhile?

To answer these questions, let us consider several numerical results which demonstrate the potential benefits of multiwavelets and shed light on results presented in this article.

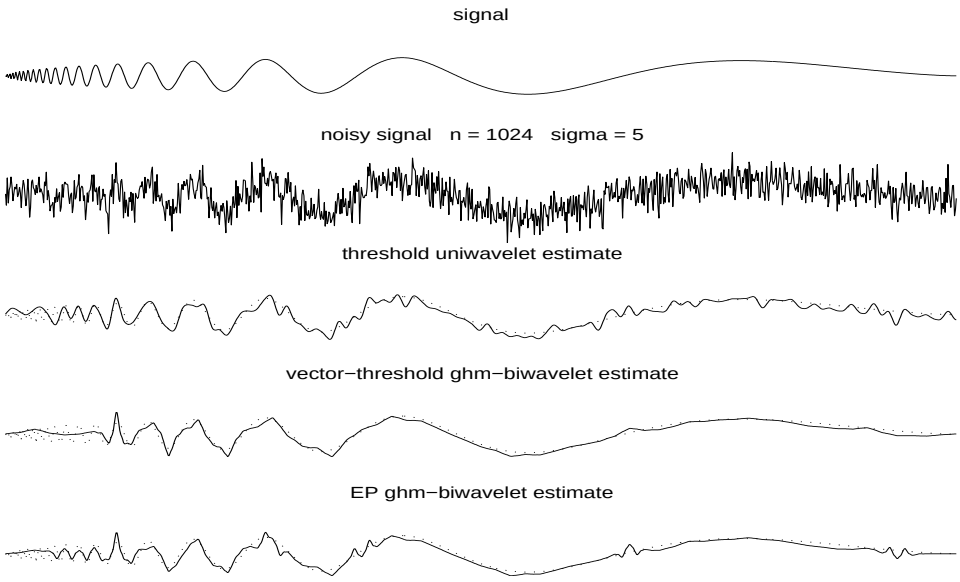

Figure 1. Denoising the doppler signal from white Gaussian noise by a universal threshold procedure for Symmlet 8 uniwavelet (the third diagram), by a vector–threshold procedure for Geronimo–Hardin–Massopust (ghm) biwavelet (the fourth diagram), and by EP denoising for ghm biwavelet (the bottom diagram). The solid and dotted lines show the denoised and underlying signals, respectively.

Figure 1 illustrates a classical denoising problem for the case of a regression model $Y_l = f(l/n) + \sigma\epsilon_l$, $l = 1, 2, \ldots, n$. Here the underlying signal f is a familiar doppler signal observed at $n = 1024$ equidistant points, $\sigma = 5$ and $\{\epsilon_1, \ldots, \epsilon_n\}$ is standard white Gaussian noise. In what follows both signals and uniwavelet procedures used are supported by the wavelet package of MATLAB. Known multiwavelet denoising procedures are supported by Strela's software available at *http://math.dartmouth.edu/strela*, and they are explained in Strela and Walden (2001).

The doppler signal, shown in the top diagram, has a challenging dynamic with a high frequency and low amplitude onset (head) and very smooth tail. The second diagram shows a noisy signal. The noise is so large that visualization of the onset of doppler is impossible. Moreover, the noise creates an illusion of high frequency oscillations in the tail, the magnitude of which is comparable to the magnitude of the oscillations in the head of the underlying doppler signal. This is a complex noisy signal, and we would like to examine how different wavelet estimators denoise the signal.

The classical threshold uniwavelet estimate (see the discussion in Mallat 1999) does a good job in demonstrating the underlying dynamic of the doppler. Also note that while the initial oscillations of the doppler signal are oversmoothed, overall the doppler's head is recovered impressively well. On the other hand, the quality of the denoising over the smooth tail can be better.

The next diagram shows how a vector–threshold denoising, which is a classical thresholding modified by Downie and Silverman (1998) to take into account the specifics of multiwavelets, filters the signal. Here the Geronimo–Hardin–Massopust (ghm) biwavelet is used. Clearly the denoising along the tail is performed much better than by the uniwavelet estimate. On the other hand, the onset of the doppler is oversmoothed. This outcome is rather typical and, in general, it is difficult to say that a multiwavelet thresholding implies a better denoising than a uniwavelet thresholding and vice versa.

The bottom diagram shows us how the Efromovich–Pinsker (EP) biwavelet estimator, developed in this article, performs. We see that this estimator combines the best features of the threshold estimates discussed earlier, but it performs a bit worse than the threshold uniwavelet estimator over the head and a bit worse than the vector–threshold multiwavelet estimator over the tail. In other words, this estimator is a reasonable compromise between the two threshold estimators.

What about data compression? In this aspect of signal denoising the EP estimate outperforms the threshold estimates. For this example the number of nonzero biwavelet coefficients of the EP estimate is just 71% and 87% of

the number of nonzero uniwavelet coefficients and vector–threshold biwavelet coefficients, respectively.

Nonetheless, despite this promising result, it is fair to conclude that for the case of white Gaussian noise there is little incentive to employ more complicated multiwavelets in place of simpler uniwavelets.

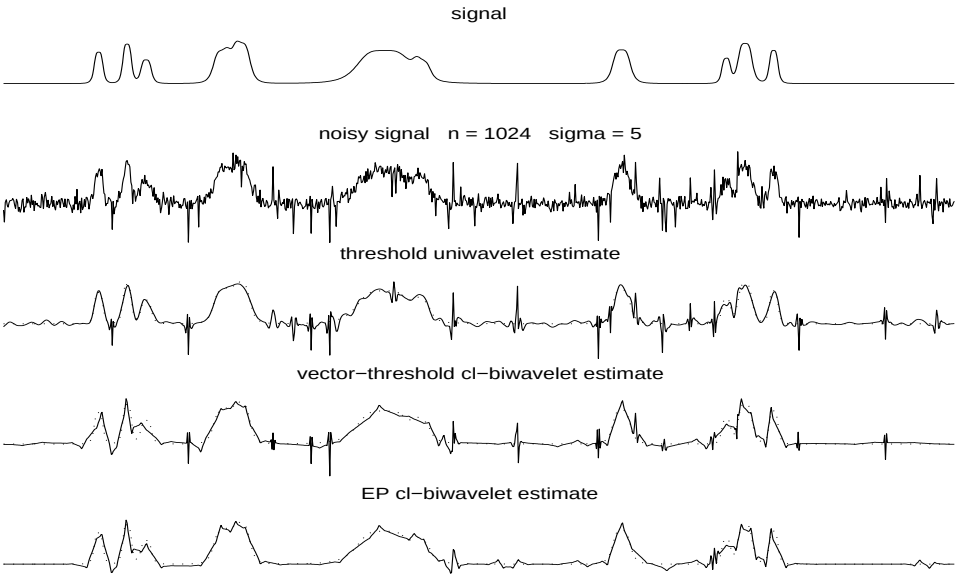

Figure 2. Denoising the bumps signal from Tukey noise by a universal threshold procedure for Symmlet 8 uniwavelet (the third diagram), by a vector–threshold procedure for Chui–Lian (cl) biwavelet (the fourth diagram), and by EP denoising for cl biwavelet (the bottom diagram). The solid and dotted lines show the denoised and underlying signals, respectively.

The situation changes dramatically if noise is not Gaussian. Figure 2 helps us to assess this issue. Here f is the familiar spatially inhomogeneous signal "bumps" that is observed in Tukey noise. Recall that a Tukey random variable is created by a mixture of two zero mean normal variables. In particular here $\epsilon = (u\zeta_1 + (1-u)4\zeta_2)/(.85)^{1/2}$ where ζ_1, ζ_2 and u are independent random variables, the first two being standard normal and the last one Bernoulli with probability $P(u = 1) = .9$. This noise is a classical example in the theory of robust estimation, see Efromovich (1999a, s.4.6).

Figure 2 indicates that neither of the threshold estimators performs an effective denoising. The estimates are contaminated by huge and confusing

noisy spikes. On the other hand, the EP biwavelet estimator does a superb job in denoising and revealing the challenging dynamics of these signals.

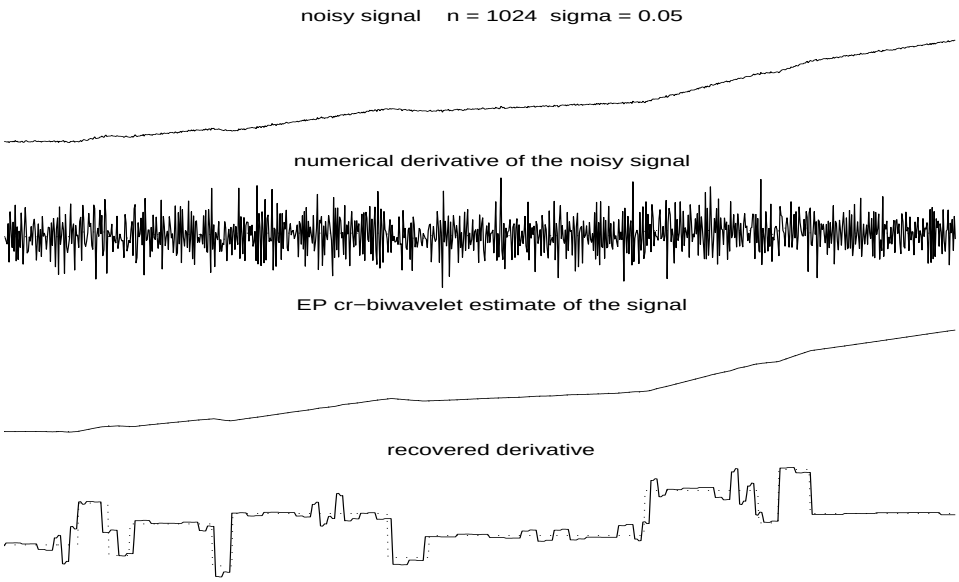

Figure 3. Recovering signal and its derivative using cristina biwavelets. The underlying derivative is the blocks signal. In the bottom diagram the solid and dotted lines show the recovered and underlying derivatives, respectively.

Now let us consider the classical problem of the simultaneous denoising a signal and the recovery of its derivative. Recall that this problem is rather simple for the case of Fourier bases and, for instance, Efromovich–Pinsker estimate performs optimal denoising and its derivative is an optimal estimate of the derivative of an underlying signal, see Efromovich (1998, 1999a, s.7.4). On the other hand, this is not the case for uniwavelets. Donoho (1995) established that no similar solution is available for uniwavelets, and that a special system of a uniwavelet and a vaguelette (biorthogonal uniwavelet) is necessary for recovering the derivative. See also the discussion in Abramovich and Silverman (1998), Cai (1999a), Johnstone (1999) and Vidakovic (1999, s.11.2). In short, uniwavelets are inevitably a more complicated tool for solving this classical inverse problem.

The situation changes dramatically if multiwavelets are used. It will be shown in Section 4 that some specific multiwavelets, in particular ones from

cristina family, allow us to solve this linear inverse problem in the same way as it is solved for Fourier bases because derivatives of the wavelet functions are again elements from the family.

To shed light on the problem of recovery of the derivative (remember that this is a classical ill-posed problem), let us consider the top diagram in Figure 3. Here again the case of equidistant regression on $[0, 1]$ is considered. The white Gaussian noise is extremely small ($\sigma = 0.05$) so there is no problem in identifying the underlying signal. But please look at the second diagram where the numerical derivative of the noisy signal is exhibited. The underlying derivative is a familiar signal in the wavelet literature, but can you recognize it? The answer is probably "no", and this is why the problem of recovering the derivative is called ill–posed because a small deviation in a signal can cause a huge deviation in its derivative.

Figure 3 illustrates how the universal EP estimator together with a cristina biwavelet performs. First, the EP estimator filters the signal, and the estimate is shown in the third diagram. The recovered derivative is shown in the bottom diagram, and it is clearly a biwavelet "miracle" because no such curve can be realized in the top diagrams. The process of obtaining this estimate is very simple and it is similar to the Fourier case. Namely to get a biwavelet coefficient for the derivative one should multiply a biwavelet coefficient of an underlying signal by a known factor that depends only on the resolution scale and then replace the biwavelet function by another from the cristina family. As a result the same software is used for denoising a signal and recovering its derivative. What we see in the bottom diagram is the realization of this procedure where the same biwavelet coefficients are used for denoising a signal and recovering its derivative but with different cristina biwavelets. Recall that threshold uniwavelet estimators use different threshold levels for denoising an underlying signal and recovering the derivative, also a special software is needed for recovering derivatives.

Let us summarize the results of our visual analysis of the figures. First of all we see that for the classical case of white Gaussian noise multiwavelets can do better than uniwavelets. This conclusion is also supported by an independent study presented in Strela and Walden (2001). However, a modest improvement in denoising does not justify the use of more complicated multiwavelets. This is probably the main reason why multiwavelets are not widely used. On the other hand, the situation changes dramatically if we "depart" from the classical Gaussian denoising model. We have seen that a relatively mild change in the noise (recall that Tukey noise is a mixture of two Gaussian noises) or solving a linear inverse problem makes using multiwavelets worthwhile. In other words, there exist special statistical problems

where multiwavelets shine, and for this problems it is worthwhile to overcome all the technical difficulties associated with using multiwavelets.

The outline of the paper is as follows. Section 2 provides a brief review of multiwavelet transforms and threshold denoising methods suggested for multiwavelets. Section 3 introduces the Efromovich–Pinsker multiwavelet estimator. Section 4 explains how this estimator is used for the recovery of the derivative. Here a new vaguelette–vaguelette approach is introduced based on the cristina family of biwavelets. This family is introduced in Lakey, Massopust and Pereyra (1998), see also Efromovich et al (1998). Asymptotic results are formulated in Section 5 and they are proved in Appendix. Note that traditional assumptions about Gaussian distribution of the noise or that it is stationary are relaxed.

2. Review of Known Multiwavelet Transforms and Estimators

We restrict our attention to biwavelets because they are the primary tool in applications. Let $\Phi(t) = (\phi^*(t), \phi^{**}(t))^T$ and $\Psi(t) = (\psi^*(t), \psi^{**}(t))^T$ be vector-columns of two scaling and two wavelet functions. Also suppose that a signal $f(t)$ may be approximated by a biwavelet expansion

$$f_J(t) = \sum_{k=0}^{n/2^J-1} S_{J,k}\Phi_{J,k}(t) + \sum_{j=1}^{J}\sum_{k=0}^{n/2^j-1} D_{j,k}\Psi_{j,k}(t), \tag{2.1}$$

where $\Phi_{j,k}(t) = 2^{-j/2}\Phi(2^{-j}x - k)$, $\Psi_{j,k}(t) = 2^{-j/2}\Psi(2^{-j}t - k)$, and the vector-rows $S_{j,k} = (s^*_{j,k}, s^{**}_{j,k})$ and $D_{j,k} = (d^*_{j,k}, d^{**}_{j,k})$ denote the corresponding wavelet coefficients. Here and in what follows we use the standard rules of multiplication of matrices. Note that each scale has two streams of wavelet functions and correspondingly two streams of wavelet coefficients.

Due to a larger flexibility, biwavelets can be more symmetric, have shorter support, and be smoother (more regular) than uniwavelets; see the discussion in Donovan *et al* (1996), Geronimo *et al* (1994), Strela and Walden (2001), Strela *et al* (1999) and Xia *et al* (1996). This together with a developed discrete biwavelet transform (DBWT), which has the same complexity $O(n)$ as its uniwavelet counterpart, implies both excellent approximation properties of biwavelets and a potential usefulness for statistical applications. On the other hand, because now for each scale two streams of wavelet coefficients should be calculated, the case of a noisy signal implies correlated and nonstationary errors in empirical wavelet coefficients. This complication is

a major challenge for statistical applications of multiwavelets. Thus let us explain it in more detail following Strela and Walden (2001).

DBWT requires an input that consists of two streams (a sequence of length-2 vectors). A procedure for creating these streams is called preprocessing, and it is a special case of matrix prefiltering. Now a bit of terminology. If a preprocessor produces n length-2 vectors it is said to be an oversampling preprocessor, while if it produces $n/2$ length-2 vectors it is called a critical sampling preprocessor. As a result, a biwavelet denoising algorithm has the following steps:

1. Prefilter the original noisy signal $\mathbf{Y} = (Y_1, Y_2, \dots, Y_n)^T$ into two streams by a specific prefilter. Note that $\mathbf{Y} = \mathbf{f} + \sigma\mathbf{e}$ where $\mathbf{f}$ and $\mathbf{e}$ are the corresponding vectors of the unobserved signal and noise. We can describe this prefiltering by a single matrix multiplication

$$\mathcal{P}\mathbf{Y} = \mathcal{P}\mathbf{f} + \sigma\mathcal{P}\mathbf{e}. \tag{2.2}$$

Then elements of this vector in odd places create the first stream and elements in even places the second one.

Clearly we have a $2n \times n$ matrix $\mathcal{P}$ for oversampling and an $n \times n$ matrix $\mathcal{P}$ for critical sampling. For instance, repeated row preprocessing is an example of oversampling and the elements of $\mathcal{P}$ are $\mathcal{P}_{ij} = 1$ if $j = 2i-1$, $\mathcal{P}_{ij} = a$ if $j = 2i$ and zero otherwise. As a result, $\mathcal{P}\mathbf{Y} = (Y_1, aY_1, Y_2, aY_2, \dots, Y_n, aY_n)^T$ and the input of DBWT is the matrix $((Y_1, aY_1)^T, \dots, (Y_n, aY_n)^T)$. Interestingly, the research by Downie and Silverman (1998) and Bui and Chen (1998) has shown that this simple prefilter is good for denoising purposes. An example of critical sampling is an $n \times n$ matrix $\mathcal{P}$ that is created by a 2×2 matrix with $\mathcal{P}_0 = ((a, b)^T, (a, -b)^T)$ placed along the diagonal and all the other elements being zero. A particular choice of constants a and b depends on the underlying biwavelets.

2. Apply DBWT. This step, made by a cascade algorithm, calculates a vector $\tilde{W}$ of empirical biwavelet coefficients. It can also be described by a single matrix multiplication,

$$\tilde{W} = \mathcal{M}\mathcal{P}\mathbf{Y} = \mathcal{M}\mathcal{P}\mathbf{f} + \sigma\mathcal{M}\mathcal{P}\mathbf{e} \stackrel{\text{def}}{=} W + \sigma\Xi. \tag{2.3}$$

Here $\mathcal{M}$ is a $2n \times 2n$ matrix for an oversampling scheme and it is $n \times n$ for a critical sampling. Matrix $\mathcal{M}$ is orthonormal for orthogonal biwavelets but not for biorthogonal ones. On the other hand, biorthogonal transformations are typically not far from orthogonal, see the discussion in Daubechies (1992, s. 8.3.5).

3. Apply a denoising procedure to the empirical biwavelet coefficients. Here the main issue is that the covariance matrix of the vector-noise Ξ is no longer diagonal; moreover, it does not correspond to a stationary process. Indeed, the covariance matrix is $\sigma^2 V$ where $V = \|V_{ij}\| = \mathcal{M}\mathcal{P}\mathcal{P}^T\mathcal{M}^T$ and the prefiltering matrix $\mathcal{P}$ is not orthogonal. A detailed numerical analysis of different covariance matrices is presented in Strela and Walden (2001). The good news is that along each stream the noise is second–order stationary and the correlation vanishes rapidly; the bad news is that for a fixed position and different streams the correlation is strong and the noise is not stationary. Based on these facts, Downie and Silverman (1998) and Strela and Walden (2001) made a conclusion that known denoising procedures developed for uniwavelets cannot be directly used for multiwavelets.

4. Apply inverse DBWT.

5. Apply postfilter and get a denoised signal.

Now we are in a position to describe two known methods of denoising supported by Strela's software.

The first method is due to Strela and Walden (2001) and it is called scalar–thresholding. The idea is to use the universal thresholding procedure modified in two aspects. First, since noise in empirical biwavelet coefficients is no longer stationary, an average variance of the noise over all biwavelet coefficients is considered. In short, the variance of the input white noise σ^2 is multiplied by the average value of the diagonal elements of the covariance matrix V. It is necessary to note that the software assumes that σ^2 is known. Second, the algorithm takes into account the number (n or $2n$) of empirical wavelet coefficients used.

The second method, called vector–thresholding, is suggested by Downie and Silverman (1998). It also mimics the universal thresholding but it takes into account a strong correlation between elements of two streams located at the same position. The underlying idea is as follows. Let Ξ_{jk}, $k = 1, \ldots, m$ be 2×1 iid random normal vectors. Denote by $A_j = E\{\Xi_{j1}\Xi_{j1}^T\}$ the covariance matrix. Then the random variables $\eta_{jk} = \Xi_{jk}^T A_j^{-1} \Xi_{jk}$ are independent and identically distributed chi–squared random variables with two degrees of freedom. Thus, as an example, for n iid random variables η_{jk} we may use the familiar relation

$$P\left(\max_{j,k} \eta_{jk} > 2\ln(n)\right) \to 0 \quad \text{as} \quad n \to \infty. \tag{2.4}$$

This relation is the mathematical justification of the following vector–threshold procedure: for all scales find the corresponding correlation matrices A_j; calculate statistics $\tilde{\eta}_{jk} = \tilde{W}_{jk}^T A_j^{-1} \tilde{W}_{jk}$; set $\hat{W}_{jk} = 0$ if $\tilde{\eta}_{jk} < \sigma^2 2\ln(n)$ and set

$\hat{W}_{jk} = \tilde{W}_{jk}$ otherwise; use the threshold empirical coefficients $\hat{W}_{jk}$ to recover an underlying signal. Note that A_j depends on the DBWT (prefilter) used and on n, so the software calculates these matrices each time as the denoising is performed.

3. EP Denoising

This procedure was suggested in Efromovich and Pinsker (1984) for a sharp optimal denoising. The underlying idea is as follows. Let $\{\varphi_i(t), i = 1, 2, \ldots\}$ be an orthonormal basis (trigonometric, wavelet, etc.) and $f(t) = \sum_{i=1}^{\infty} \theta_i \varphi_i(t)$ where θ_i are the corresponding coefficients that can be estimated by empirical coefficients $\tilde{\theta}_i$. It is assumed that the empirical coefficients are unbiased estimates, i.e., $E\{\tilde{\theta}_i\} = \theta_i$.

The notion of an "oracle" is familiar in the curve estimation literature and it is a procedure that knows both data and the underlying signal f; the discussion of different types of oracles can be found in Efromovich (1999a, s.3.2). Let us consider a block smoothing oracle

$$\tilde{f}^*(t) = \sum_{m=1}^{\infty} \lambda_m \sum_{i \in T_m} \tilde{\theta}_i \varphi_i(t). \tag{3.1}$$

Here the blocks T_m are consecutive and nonoverlaping blocks of positive integer numbers, and the weights $0 \le \lambda_m \le 1$ may depend on f. Thus, the oracle can use the information about an underlying signal to choose optimal weights. Then a simple algebra shows that an optimal weight λ_m^* that minimizes the mean squared error $E\{\sum_{i \in T_m} (\lambda_m \tilde{\theta}_i - \theta_i)^2\}$ (and thus by Parseval identity it also minimizes the mean integrated squared error of $\tilde{f}^*$) is

$$\lambda_m^* = \frac{\sum_{i \in T_m} \theta_i^2}{\sum_{i \in T_m} \theta_i^2 + \sum_{i \in T_m} \mathrm{Var}(\tilde{\theta}_i)}. \tag{3.2}$$

EP denoising is based on direct mimicking (3.2). Assume for a moment that the variances $\mathrm{Var}(\tilde{\theta}_j)$ are known. Then the only unknown functional of f in (3.2) is the sum of the squared coefficients $\sum_{i \in T_m} \theta_i^2$. An unbiased estimate of this functional is $\sum_{i \in T_m} (\tilde{\theta}_i^2 - \mathrm{Var}(\tilde{\theta}_i))$, and this is the estimate used. Also note that the functional is nonnegative so a thresholding should be used to get a nonnegative estimate. In particular, using a hard thresholding yields

$$\hat{\lambda}_m = \frac{\sum_{i \in T_m} \tilde{\theta}_i^2 - \sum_{i \in T_m} \mathrm{Var}(\tilde{\theta}_i)}{\sum_{i \in T_m} \tilde{\theta}_i^2} I\left(\sum_{i \in T_m} \tilde{\theta}_i^2 > c_m \sum_{i \in T_m} \mathrm{Var}(\tilde{\theta}_i) \right), \tag{3.3}$$

where c_m may depend on n and $I(\cdot)$ is the indicator function. Note that a smooth thresholding can be used as well.

This estimator was introduced in Efromovich and Pinsker (1984); see also the discussion in Efromovich (1999a, ch.7, 1999b). Note that λ^*_m depends only on variance of the underlying noise but not on its distribution. This explains robustness of Efromovich–Pinsker estimator.

Now let us describe a particular EP denoising procedure suggested for multiwavelets and sample sizes up to several thousands.

Six resolution scales are used in (2.1), that is, $J = 6$. For the scale functions and the two coarsest scales of wavelet functions, the length of blocks is 1, i.e., each coefficient of these scales is smoothed individually. Also for the coarsest scales the coefficients c_m are all the same and equal to 1.

For the finer scales blocks are increased. Namely, for the fourth scale blocks are the "single" arrays $(d^*_{4,m}, d^{**}_{4,m})$, $m = 0, \dots, n/2^4 - 1$. For the third scale the blocks include two adjoint single arrays, for the second scale they include three adjoint single arrays, and for the finest scale they include four adjoint single arrays. For these scales $c_m = 5$.

Let us explain how the variances $\mathrm{Var}(\tilde{\theta}_i)$ are estimated. Recall that a method used should be self–learning, that is, no information about prefilter and DBWT should be required. In other words, the EP estimator considers the software as a "black–box".

Let us assume for a moment that the scale parameter σ of the noise is known. Then the following Monte Carlo learning procedure is used. A standard white Gaussian noise is used as an input signal. Then for each scale (i.e., each j) the corresponding empirical wavelet coefficients $\tilde{d}^*_{j,k}$, $k = 0, \dots, n/2^j - 1$ are second order stationary, and the same is true for $\tilde{d}^{**}_{j,k}$, $k = 0, \dots, n/2^j - 1$. Thus, a sample variance estimator can be used. Then these estimates are multiplied by σ^2 and this gives us estimates of $\mathrm{Var}(\tilde{d}^*_{jl})$ and $\mathrm{Var}(\tilde{d}^{**}_{jl})$ respectively. Then this learning procedure is repeated m_0 times (the m_0 used in applications is 7) and the results are averaged. This ends the learning part.

To assess the accuracy of this method, recall that Strela's software allows us to calculate these variances exactly. Numerical experiments have revealed that the suggested learning procedure yields a sufficient accuracy of estimation of the variances. Also note that by increasing m_0 any desired accuracy can be achieved since the classical Monte Carlo approach is used.

We conclude that for statistical applications it is not necessary to know a particular type of prefilter or DBWT used. For an applied statistician this

makes multiwavelet software similar to uniwavelet software. Also note that since only variances are necessary for the EP denoising, no calculation of $n \times n$ or $2n \times 2n$ covariance matrices is required.

Finally, let us explain how the scale parameter σ is estimated. Any robust method known for uniwavelets can be used. All these methods are based on analyzing wavelet coefficients on the finest scale. For a multiwavelet the only difference is that multiwavelet coefficients should be rescaled using results of the previous learning process. Then, if an underlying signal is zero, the variance of the rescaled empirical coefficients is the wished σ^2.

Several classical methods of estimating σ were tested. The first was the classical MAD used by all commercial wavelet softwares. This procedure worked perfectly for Gaussian noise but worse for other types of noise. Two other methods were the naive sample standard deviation estimator and the difference method. These two methods are well known in the regression literature, see Efromovich (1999a, ch.4). Numerical experiments revealed that for normal noise and $\sigma > .5$ the MAD and sample standard deviation methods performed similarly. For $\sigma < .5$ the MAD performed essentially better, and this was not a surprise. For non-Gaussian noise, such as Tukey noise, the sample standard deviation performed better than MAD. Interestingly, the overall winner of this limited competition was the difference method—it consistently produced robust and accurate estimation of σ.

Table 1. Sample means of ratios of ISE and data compression between EP estimates and uniwavelet (vector–threshold) estimates. Each element in the body of the table is written as $\frac{A}{B}$ where: A is the sample mean of ratios for EP estimator and uniwavelet estimator; B is the sample mean of ratios for EP estimator and vector–threshold estimator.

n	512			1024			2048		
σ	2	5	8	2	5	8	2	5	8
ISE									
"bumps"	$\frac{0.74}{0.85}$	$\frac{0.58}{0.54}$	$\frac{0.52}{0.59}$	$\frac{0.77}{0.86}$	$\frac{0.65}{0.67}$	$\frac{0.58}{0.68}$	$\frac{0.63}{0.76}$	$\frac{0.68}{0.71}$	$\frac{0.78}{0.75}$
"doppler"	$\frac{0.81}{1.03}$	$\frac{0.94}{0.98}$	$\frac{0.95}{0.98}$	$\frac{0.92}{1.01}$	$\frac{0.87}{0.98}$	$\frac{0.92}{0.94}$	$\frac{0.86}{0.91}$	$\frac{0.87}{0.98}$	$\frac{0.95}{0.97}$
"blocks"	$\frac{0.77}{0.75}$	$\frac{0.65}{0.71}$	$\frac{0.71}{0.75}$	$\frac{0.68}{0.57}$	$\frac{0.62}{0.58}$	$\frac{0.64}{0.61}$	$\frac{0.53}{0.58}$	$\frac{0.61}{0.58}$	$\frac{0.58}{0.57}$
Data Compression									
"bumps"	$\frac{0.43}{0.80}$	$\frac{0.51}{0.78}$	$\frac{0.58}{0.82}$	$\frac{0.64}{0.73}$	$\frac{0.62}{0.79}$	$\frac{0.68}{0.76}$	$\frac{0.65}{0.78}$	$\frac{0.70}{0.77}$	$\frac{0.67}{0.79}$
"doppler"	$\frac{0.49}{0.61}$	$\frac{0.52}{0.77}$	$\frac{0.75}{0.87}$	$\frac{0.73}{0.92}$	$\frac{0.74}{0.86}$	$\frac{0.72}{0.85}$	$\frac{0.69}{0.87}$	$\frac{0.68}{0.78}$	$\frac{0.64}{0.79}$
"blocks"	$\frac{0.68}{0.71}$	$\frac{0.69}{0.87}$	$\frac{0.85}{0.88}$	$\frac{0.86}{0.92}$	$\frac{0.83}{0.92}$	$\frac{0.79}{0.89}$	$\frac{0.76}{0.85}$	$\frac{0.81}{0.91}$	$\frac{0.83}{0.92}$

Figure 1 showed us just one particular example of how the estimator performed. A similar intensive Monte Carlo study was conducted as well. Define an experiment as a combination of an underlying signal, sample size

n, and standard deviation σ. For every experiment: (i) 300 independent Monte Carlo simulations were made; (ii) the sample means were calculated over these 300 simulations for each of the ratios: the ISE of the EP estimator / the ISE of the uniwavelet estimator, and the ISE of the EP estimator / the ISE of vector–threshold multiwavelet estimator; (iii) the sample means (over these 300 simulations) were calculated for the ratios: number of non-zero wavelet coefficients used by EP estimator /number of non-zero wavelet coefficients used by uniwavelet estimator, and number of non-zero wavelet coefficients used by EP estimator /number of non-zero wavelet coefficients used by vector–threshold multiwavelet estimator.

Step (ii) allows us to compare the estimates in terms of ISE whereas step (iii) allows us to compare the data compression properties of the estimates. Note that if the ratio is smaller than 1 then the EP estimator is better than the other estimates and vice versa. The results are presented in Table 1. There is also an interesting byproduct of this table. The ratio $\frac{A}{B}$ allows us to compare the uniwavelet and multiwavelet threshold estimators. If this ratio is less than 1 then the multiwavelet threshold estimator is better than the uniwavelet one and vice versa.

The results show that in terms of accuracy of denoising, the EP estimator significantly outperforms the threshold estimators for blocks and bumps signals, and it performs better for the doppler signal. In terms of data compression, the EP estimator significantly outperforms the threshold estimators. There is no clear winner between uniwavelet and biwavelet threshold estimators. This conclusion is supported by other known numerical studies as well. It explains that, to compete with uniwavelets, special multiwavelet denoising procedure should be developed.

Keeping in mind that normal noise favors the threshold uniwavelet estimator (recall the discussion in Introduction), these numerical results are very promising. Moreover, even for the case of a normal noise, the results may justify the additional efforts involved in the using multiwavelets.

4. **Recovery of the Derivative**

A classical filtering problem includes denoising a signal and estimation of different functionals, see Efromovich (1999a, s.7.1). One of most frequently considered functionals is the derivative of an underlying signal. This is also one of the most complicated functionals because the setting is ill-posed. Recall that we briefly discussed this issue in Introduction.

Our setting is as follows. The problem is to estimate the derivative $f^{(1)}(t) = df(t)/dt$ of an underlying signal f for the model of an equidistant regression on $[0, 1]$. At first glance, a possible solution looks straightforward. Assume that an underlying signal can be written as $f(t) = \sum_{jk} \theta_{j,k}\psi_{j,k}(t)$, $0 \le t \le 1$ where the sum is over a finite set of the indexes. If $\psi_{j,k}(t)$ are differentiable then $f^{(1)}(t) = \sum_{jk} \theta_{j,k}\psi^{(1)}_{j,k}(t)$. Thus it is natural to assume that one can just differentiate a good estimate of f to get a good estimate of the derivative and then replace $\{\psi^{(1)}_{j,k}\}$ by the corresponding linear combinations of $\{\psi_{j,k}\}$. Interestingly, under mild assumptions this naive approach leads to optimal estimates when a Fourier basis is used, see Efromovich (1998, 1999a, s.7.1-3). This nice feature of Fourier bases is explained by the fact that derivatives of its elements (sine and cosine functions) are again elements of the bases. In particular, this makes the EP Fourier estimator universal because the estimator and its derivatives are optimal estimates of an underlying signal and its derivatives.

On the other hand, it is well known that threshold uniwavelet estimates are not universal and, moreover, serious problems arise when such an approach is used. First of all, Donoho (1995) established that $\psi^{(1)}_{j,k}$ is no longer a uniwavelet but (under mild assumptions) a univaguelette, i.e., a biorthogonal uniwavelet. Thus a signal and its derivative should be approximated using elements of different wavelet families, and a discrete wavelet transform (DWT) used for denoising an underlying signal no longer can be used for recovery of the derivative which is expanded via vaguelettes. Instead, an inverse vaguelette discrete transform is used. The approach when one first expands f via wavelets and then restores the derivative via vaguelettes is called a vaguelette–wavelet decomposition. Of course it is possible to do vice versa. Namely, one can expand $f^{(1)}(t)$ via wavelets and f via vaguelettes. This approach is called wavelet–vaguelette decomposition. The interested reader can find more about these methods in Donoho (1995), Abramovich and Silverman (1998), Johnstone (1999) and Cai (1999a).

Another serious complication is that different threshold levels should be used for denoising a signal and recovery of its derivative whenever a threshold method is employed. This is not difficult to understand keeping in mind that $\psi^{(1)}_{j,k}(t) = 2^{3j/2}\psi^{(1)}(2^j t - k)$, see the discussion in the above–mentioned articles.

A natural question is as follows. Is it possible to suggest a wavelet estimator that allows us to solve the problem of a simultaneous estimation of a function and its derivative similarly to the Fourier case? The answer is "yes" and the approach suggested may be called a vaguelette–vaguelette

decomposition because a recommended universal EP estimator is based on cristina family of biorthogonal biwavelets. The important property of these biwavelets is that if the expansion (2.1) holds then the derivative $f_J^{(1)}(t)$ of $f_J(t)$ can be written as

$$f_J^{(1)}(t) = \sum_{k=0}^{n/2^J-1} 2^{-J} S_{J,k} T \Phi^-_{J,k}(t) + \sum_{j=1}^{J} \sum_{k=0}^{n/2^j-1} 2^{-j} D_{j,k}(-1) \Psi^-_{j,k}(t), \quad (4.1)$$

where $\Psi^-_{j,k}$ is a so-called roughened biwavelet function from cristina family and T is a shift-type operator applied to the roughened scale vector–function $\Phi^-_{J,k}(t)$; see details in Lakey, Massopust and Pereyra (1998).

This family of biwavelets together with Efromovich–Pinsker denoising allows us to perform a simultaneous estimation of a signal and its derivative. In short, to estimate a signal or its derivative a corresponding cristina filter bank is used at the step of reconstruction. As a result, the important difference between this multiwavelet estimator and the known uniwavelet methods is that neither a special denoising procedure nor new software is required.

Finally, recall that Figure 3 illustrates the performance of EP estimator based on cristina biwavelets. Keeping in mind that we discuss an ill-posed problem, the multiwavelet estimator performs a miracle because no visual analysis helps in revealing the underlying derivatives. More examples are presented in Efromovich *et al* (1998).

5. Asymptotic Properties of EP Multiwavelet Estimator

In this section we consider a traditionally studied model of equidistant regression on $[0, 1]$

$$Y_l = f(l/n) + \epsilon_l, \quad l = 1, 2, \ldots, n, \quad (5.1)$$

where ϵ_l, $l = 1, \ldots, n$ are independent zero mean and unit variance random variables that may have different distributions (i.e., they are a second order white noise).

Using a traditional in the wavelet literature approach, we begin with the analysis of oracles and establishing oracle inequalities. Then an adaptive estimator will be introduced and compared with the oracles.

One of the main attractive properties of wavelets is that they have excellent approximation and data compression properties for functions with

spikes and jumps on a smooth background. Following the pioneering results of Hall, Kerkyacharian and Picard (1998,1999), let us introduce a Hall–Kerkyacharian–Picard $\mathrm{HKP}_\alpha \subset L_2([0,1])$ function class. A function from this class is the sum of a smooth (regular) function g_1 from a Besov space $B^\alpha_{\infty\infty}$ and an irregular function g_2. The g_2 can be either a piecewise polynomial with a finite number of discontinuities or a function of the form $\sum_{l=1}^r A_l(x-x_l)^{a_l}\cos(x-x_l)^{-b_l}$. The latter functions include the "chirp" and "doppler" signals familiar in the wavelet literature , and overall the HKP_α class includes all classical signals used in the wavelet literature. Then assuming that a uniwavelet has a finite support, is sufficiently regular, $\alpha > 1/2$ and some other mild assumptions, it was established that for a large class of distributions of the noise in (5.1) the uniwavelet block threshold oracle

$$\tilde f_n^*(t) = \sum_{k=0}^{2^{j_0}-1} \hat s_{j_0,k}\phi_{j_0,k}(t)$$

$$+ \sum_{j=j_0}^{J^*}\sum_{m=1}^{2^j/L_{jn}} I(L_{jn}^{-1}\sum_{k\in T_{jmn}} d_{j,k}^2 > c^* n^{-1}) \sum_{k\in T_{jmn}} \hat d_{j,k}\psi_{j,k}(t) \qquad (5.2)$$

is rate optimal, that is,

$$\sup_{f\in \mathrm{HKP}_\alpha} E\{\int_0^1 (\tilde f_n^*(t) - f(t))^2 dt\} \le C n^{-2\alpha/(2\alpha+1)}. \qquad (5.3)$$

In this section $\phi_{j,k}(t) = 2^{j/2}\phi(2^j t - k)$, $\psi_{j,k}(t) = 2^{j/2}\psi(2^j t - k)$ and, for simplicity in exposition, it is assumed that wavelet bases are periodized on $[0,1]$. Also in (5.2) it is assumed that $2^{j_0-1} \le n^{1/(2N+1)} \le 2^{j_0}$ and N is the wavelet regularity; $T_{jmn} = \{k:\ L_{jn}(m-1) \le k < L_{jn}m\}$ are the blocks at jth resolution scale and the blocks may depend on n; L_{jn} is the length of blocks at the jth scale and it is assumed that $2^j/L_{jn}$ is integer; J^* is the rounded down $\log_2(n/\ln(n))$; c^* is a positive constant. Here and in what follows C's are generic finite constants, and recall that $I(\cdot)$ is the indicator.

There is a wide variety of blocks that imply the rate optimality; for instance, blocks with logarithmic length imply very attractive global and pointwise properties (see Cai 1999b and Efromovich 1999a, s.7.4). In what follows we are concentrating on conditions that imply smallest blocks because this allows us to utilize the best features of multiwavelets—short support and large number of vanishing moments.

It is necessary to note that a class of functions for which (5.3) holds is essentially larger than the particular HKP_α class; see, for instance, Cai

(1999b), Hall, Kerkyacharian and Picard (1998,1999). Thus, with some abuse of notation, we shall refer to a class of functions such that (5.3) holds as HKP_α.

The procedure (5.2) is a block threshold oracle because it keeps empirical wavelet coefficients from a block only if the underlying wavelet coefficients are sufficiently large. It is well known that the block threshold oracle is dominated by an Efromovich–Pinsker (EP) oracle with weights (3.2). This assertion follows from the inequality

$$E\{(\lambda(\theta+\epsilon)-\theta)^2\} \geq E\{(\lambda^*(\theta+\epsilon)-\theta)^2\} \quad \text{where} \quad \lambda^* = \theta^2/(\theta^2+\mathrm{Var}(\epsilon)),$$

that holds for any zero mean and finite–variance random variable ϵ. The interested reader can find a discussion about this and other oracles used in nonparametric curve estimation in Efromovich (1999a, s. 3.2).

Proof of (5.3) is based only on the assumption of a compact support and a sufficient regularity of a uniwavelet used. Fulfilling these properties is not an issue for biwavelets, thus (5.3) holds for multiwavelets as well.

In what follows we simultaneously consider orthogonal and biorthogonal biwavelets. Recall that biorthogonal biwavelets form a Riesz basis and thus they are "almost" orthogonal. In other words, $c_1\|f\|^2 \leq \int f^2(t)dt \leq c_2\|f\|^2$ where c_1 and c_2 are positive constants and here and in what follows $\|f\|^2$ is the sum of squared wavelet coefficients (squared norm in l_2). For orthonormal wavelets we have $c_1 = c_2 = 1$ by Parseval identity. Cristina biorthogonal biwavelets, defined in Lakey, Massopust and Pereyra (1998) and used in the examples, have $c_2 < 1.27$ and $c_1 > .81$, and thus they are indeed almost orthonormal.

Let us formulate assumptions about prefilter, DBWT and noise. Recall that the covariance matrix $\|V_{ij}\| = \mathcal{M}\mathcal{P}\mathcal{P}^T\mathcal{M}^T$ of errors in empirical biwavelet coefficients was introduced in Section 2. Also denote by $v_j = (2L_{jn})^{-1}n\sum_{k\in T_{jmn}}[\mathrm{Var}(\hat{d}^*_{j,k}) + \mathrm{Var}(\hat{d}^{**}_{j,k})]$ the average (over the block T_{jmn}) variance of the noise.

ASSUMPTION 1. A prefilter and a DBWT used are such that for all n the eigenvalues of the covariance matrix $\|V_{ij}\|$ are bounded by a finite λ^*, $V_{jj} \leq v^* < \infty$ and $v_j \geq v_* > 0$.

ASSUMPTION 2. Errors $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$ in (5.1) are zero mean and unit variance independent random variables. Two possible cases of their distributions are considered. The first one is where the distribution is normal and thus all these errors are iid standard normal. The second case is where the distributions can be different but for all vectors $(b_1, b_2, \ldots, b_n)$ of the unit

length, i.e., $\sum_{l=1}^n b_l^2 = 1$, and some positive t_0

$$E\{\exp(t_0[\sum_{l=1}^n b_l\epsilon_l]^2)\} \le C. \tag{5.4}$$

In the second case it is also assumed that the noise in empirical biwavelet coefficients is m–dependent along a stream of the coefficients.

All prefilters and DBWT discussed in Strela and Walden (2001), as well as the ones used for cristina biwavelets, satisfy Assumption 1. Let us also comment on Assumption 2. The first case is the classical white Gaussian noise. It allows us to compare the EP biwavelet estimator with best adaptive uniwavelet estimators because this is the model that is traditionally considered. The second case allows us to consider a broader class of noise with exponential moments. This class includes Tukey noise discussed in Introduction. The assumption on m–dependence requires using biwavelets with finite support, see Neumann and Spokoiny (1995). Also note that we consider errors with finite exponential moments only because we explore smallest blocks, see the discussion in Efromovich (1999a, s.7.4).

These assumptions together with Hall, Kerkyacharian and Picard (1999) imply that the EP biwavelet oracle

$$\hat f_n^*(t) = \sum_{k=0}^{2^{j_0}-1} \hat S_{j_0,k}\Phi_{j_0,k}(t) + \sum_{j=j_0}^{J^*}\sum_{m=1}^{2^j/L_{jn}}\sum_{k\in T_{jmn}} D_{j,k}D_{j,k}^T$$

$$\times \left(\sum_{k\in T_{jmn}} D_{j,k}D_{j,k}^T + 2L_{jn}v_j n^{-1}\right)^{-1} \sum_{k\in T_{jmn}} \hat D_{j,k}\Psi_{j,k}(t) \tag{5.5}$$

is rate minimax, that is,

$$\sup_{f\in \mathrm{HKP}_\alpha} E\{\int_0^1 (\hat f_n^*(t) - f(t))^2 dt\} \le Cn^{-2\alpha/(2\alpha+1)}. \tag{5.6}$$

Moreover, according to Efromovich(1999a, s.3.2, s.7.4), this oracle dominates all other known oracles used in the wavelet literature.

Now consider a problem of recovery of the derivative $f^{(1)}$ of a signal f. For this problem it is assumed that a cristina biwavelet is used. In this case, similarly to a Fourier basis, $\hat f^{*(1)}$ is an EP oracle for estimation of the derivative $f^{(1)}$, see the discussion in Efromovich(1999a, s.7.4). Thus

the oracle is universal. This together with a simple algebra shows that for $\alpha > 1.5$

$$\sup_{f\in \mathrm{HKP}_\alpha} E\{\int_0^1 (\hat{f}_n^{*(1)}(t) - f^{(1)}(t))^2 dt\} \le Cn^{-2(\alpha-1)/(2\alpha+1)}. \tag{5.7}$$

This concludes our discussion of oracles. Now we are in a position to formulate a proposition on how well an EP adaptive estimator mimics these oracles.

Define a biwavelet EP adaptive estimator

$$\hat{f}_n(t) = \sum_{k=0}^{2^{j_0}-1} \hat{S}_{j_0,k}\Phi_{j_0,k}(t) + \sum_{j=j_0}^{J^*}\sum_{m=1}^{2^j/L_{jn}} \frac{(2L_{jn})^{-1}\sum_{k\in T_{jmn}} \hat{D}_{j,k}\hat{D}_{j,k}^T - v_j n^{-1}}{(2L_{jn}^{-1})\sum_{k\in T_{jmn}} \hat{D}_{j,k}\hat{D}_{j,k}^T}$$

$$\times I\left((2L_{jn}^{-1})\sum_{k\in T_{jmn}} \hat{D}_{j,k}\hat{D}_{j,k}^T - v_j n^{-1} > c_{jn}v_j n^{-1}\right)\sum_{k\in T_{jmn}} \hat{D}_{j,k}\Psi_{jk}(t). \tag{5.8}$$

Denote $f^{(0)} = f$ and recall that C's are generic positive constants.

THEOREM 1. *Consider model (5.1). Let Assumptions 1–2 hold and* $\nu \in \{0,1\}$. *Also assume that parameters of the EP estimator satisfy* $(L_{jn}c_{jn}^2)^{-1} + c_{jn} < d_n$ *and*

$$\sum_{j=j_0}^{J^*} 2^{j(1+2\nu)} L_{jn}^{-1/2} \exp\{-\kappa L_{jn}c_{jn}^2\} \le d_n' n E\{\|\hat{f}_n^{*(\nu)} - f^{(\nu)}\|^2\}, \tag{5.9}$$

where $d_n < C$, $d_n' < C$ *and* $\kappa = \kappa(t_0, \lambda^*, v^*, v_*) > 0$. *Then mean integrated squared errors of the EP adaptive biwavelet estimator and its derivative are within a factor from the mean integrated squared errors of the corresponding oracles* $\hat{f}_n^{*(\nu)}$, *that is,*

$$E\Big\{\|\hat{f}_n^{(\nu)} - f^{(\nu)}\|^2\Big\} \le CE\Big\{\|\hat{f}_n^{*(\nu)} - f^{(\nu)}\|^2\Big\}. \tag{5.10}$$

Moreover, if additionally $d_n = o(1)$ *and* $d_n' = o(1)$ *then a sharp mimicking occurs, that is,*

$$E\Big\{\|\hat{f}_n^{(\nu)} - f^{(\nu)}\|^2\Big\} \le (1 + o(1))E\Big\{\|\hat{f}_n^{*(\nu)} - f^{(\nu)}\|^2\Big\}. \tag{5.11}$$

Note that $nE\{\|\hat{f}_n^{*(\nu)} - f^{(\nu)}\|^2\} > C > 0$ whenever f is not a constant, then (5.9) is valid if its left side vanishes as $n \to \infty$.

We have established that asymptotically the data–driven EP biwavelet estimator performs as well as the oracles and its derivative also mimics the corresponding oracle. In other words, the biwavelet estimator is universal following the terminology of Efromovich (1999a, ch.7). Also note that the proposition implies that there exists a whole family of blocks and threshold levels for which the EP estimator is universal. This is an important corollary for statistical applications.

Theorem 1 together with (5.6)–(5.7) also implies that if $(L_{jn}c_{jn}^2)^{-1} + c_{jn} \le C$ and

$$\sum_{j=j_0}^{J^*} 2^{j(1+2\nu)} L_{jn}^{-1/2} \exp\{-\kappa L_{jn} c_{jn}^2\} \le C n^{(1+2\nu)/(2\alpha+1)}, \tag{5.12}$$

then for $\alpha > \nu + 1/2$

$$\sup_{f \in \mathrm{HKP}_\alpha} E\{\|\hat{f}_n^{(\nu)} - f^{(\nu)}\|^2 \le C n^{-2(\alpha-\nu)/(2\alpha+1)}. \tag{5.13}$$

This corollary yields both the minimaxity and minimax universality of the adaptive biwavelet estimator.

There is one more interesting corollary. It follows from (5.9) that it is possible to choose blocks of a logarithmic length. According to Cai (1999b), for the case of normal errors and uniwavelets this fact implies pointwise optimality of the adaptive estimate, that is, its mean squared error converges with optimal adaptive rate. It is possible to verify that the same result holds for biwavelets as well, the proof will be presented elsewhere.

Finally recall that the case of errors with exponential moments has been considered only to indicate blocks of minimal lengths. It is possible to relax this assumption and consider errors with, for instance, finite eighth moments, but these errors will require larger blocks for optimal denoising. In short, there is a tradeoff between the assumption on moments of errors and length of blocks used, see also the discussion in Efromovich (1999a, s.7.4).

Conclusion

Multiwavelets have superior approximation and data compression properties, they can have shorter support, be more symmetric and be smoother than uniwavelets. On the other hand, using multiwavelets for signal denoising implies serious technical complications due to nonstationary noise in empirical multiwavelets coefficients. This fact, combined with known

numerical experiments showing that threshold multiwavelet denoising does not always outperform threshold uniwavelet denoising, explains why multiwavelets are rarely used in statistical applications. This article shows that multiwavelets become an attractive tool whenever more complicated statistical problems are considered, in particular a denoising from a non-Gaussian noise and solving inverse problems. An adaptive and self-learning Efromovich–Pinsker estimator is suggested that requires no knowledge about the prefilter–DBWT used and about smoothness of an underlying signal. Asymptotically this adaptive estimator mimics the performance of wavelet oracles. The estimator, used with a cristina biwavelet, is also universal meaning that its derivative mimics the performance of the corresponding oracle. The approach, suggested for the recovery of the derivative, is based on a vaguelette–vaguelette decomposition and it resembles a classical singular value decomposition method used for Fourier bases. Numerical simulations show that the estimator performs exceptionally well in terms of accuracy of denoising and data compression. The results indicate that multiwavelets can be an attractive tool in statistical applications.

Appendix

PROOF OF THEOREM 1. Let us simplify notation. Set

$$\hat{\Theta}_{jm} = (2L_{jn})^{-1} \sum_{k \in T_{jmn}} \hat{D}_{j,k}\hat{D}_{j,k}^T - v_j n^{-1},$$

$$\hat{\Lambda}_{jm} = \frac{\hat{\Theta}_{jm}}{\hat{\Theta}_{jm} + v_j n^{-1}} I(\hat{\Theta}_{jm} > c_{jn} v_j n^{-1}),$$

$$\Theta_{jm} = (2L_{jn})^{-1} \sum_{k \in T_{jmn}} D_{j,k}D_{j,k}^T, \quad \Lambda_{jm}^* = \frac{\Theta_{jm}}{\Theta_{jm} + v_j n^{-1}}.$$

Also we omit indices and limits in sums whenever no confusion occurs.

Write

$$A \stackrel{\text{def}}{=} E\{\|(\hat{f}_n^{(\nu)} - \hat{f}_n^{*(\nu)}\|^2\} = \sum_{j=j_0}^{J^*} \sum_{m=1}^{2^j/L_{jn}} 2^{2j\nu} E\{(\hat{\Lambda}_{jm} - \Lambda_{jm}^*)^2 (2L_{jn})(\hat{\Theta}_{jm} + v_j n^{-1})\}.$$

The weight $\hat{\Lambda}_{jm}$ is either positive or zero. The latter occurs when empirical biwavelet coefficients from the block T_{jmn} are not large enough. We

are considering these two cases separately. Write

$$A = \sum_j \sum_m 2^{2j\nu}(2L_{jn})E\{v_j^2 n^{-2}(\hat\Theta_{jm}-\Theta_{jm})^2 I(\hat\Theta_{jm}\ge c_{jn}v_j n^{-1})$$
$$\times[(\hat\Theta_{jm}+v_j n^{-1})(\Theta_{jm}+v_j n^{-1})^2]^{-1}\}$$
$$+\sum_j \sum_m 2^{2j\nu}(2L_{jn})(\Lambda^*_{jm})^2 E\{(\hat\Theta_{jm}+v_j n^{-1})I(c_{jn}v_j n^{-1}>\hat\Theta_{jm})\}$$
$$\stackrel{\text{def}}{=} A_1 + A_2. \tag{A.1}$$

Now we are estimating these two terms. Let us begin with A_1.

$$A_1 = n^{-2}\sum_j \sum_m 2^{2j\nu}(2L_{jn})E\{v_j^2(\hat\Theta_{jm}-\Theta_{jm})^2(\hat\Theta_{jm}+v_j n^{-1})^{-1}I(\hat\Theta_{jm}>c_{jn}v_j n^{-1})\}$$
$$\times(\Theta_{jm}+v_j n^{-1})^{-2}[I(\Theta_{jm}\ge c_{jn}v_j n^{-1}/2)+I(c_{jn}v_j n^{-1}/2>\Theta_{jm})].$$

We need the following technical result (it will be proved later).

LEMMA 1. *If Assumptions 1 and 2 hold then*

$$E\{(\hat\Theta_{jm}-\Theta_{jm})^2\}\le CL_{jn}^{-1}v_j n^{-1}(\Theta_{jm}+v_j n^{-1}), \tag{A.2}$$

and there exists a positive $\kappa=\kappa(t_0,\lambda^*,v^*,v_*)$ *such that*

$$P(\hat\Theta_{jm}>c_{jn}v_j n^{-1},\Theta_{jm}<c_{jn}v_j n^{-1}/2)\le C\exp\{-\kappa L_{jn}c_{jn}^2\}. \tag{A.3}$$

Let us continue the estimation of A_1. Note that $\Lambda^*_{jm}\ge(1+2/c_{jn})^{-1}$ whenever $\Theta_{jm}\ge c_{jn}v_j n^{-1}/2$. Using this fact, Assumption 1, (A.2), (A.3) and Cauchy–Schwarz inequality we get

$$A_1 \le Cn^{-1}\sum_j\sum_m 2^{2j\nu}L_{jn}v_j\Lambda^*_{jm}(1+c_{jn}^{-1})v_j n^{-1}L_{jn}^{-1}(\Theta_{jm}+v_j n^{-1})$$
$$\times(\Theta_{jm}+v_j n^{-1})^{-2}I(\Theta_{jm}>c_{jn}v_j n^{-1}/2)$$
$$+Cn^{-1}\sum_j\sum_m 2^{2j\nu}L_{jn}v_j[L_{jn}^{-1}v_j n^{-1}(\Theta_{jm}+v_j n^{-1})\exp\{-\kappa L_{jn}c_{jn}^2\}]^{1/2}$$
$$\times I(\Theta_{jm}<c_{jm}v_j n^{-1}/2)(\Theta_{jm}+v_j n^{-1})^{-1}$$
$$\le Cn^{-1}\sum_{j=j_0}^{J^*}\sum_{m=1}^{2^j/L_{jn}}2^{2j\nu}v_j[L_{jn}\Lambda^*_{jm}(L_{jn}c_{jn})^{-1}+L_{jn}^{1/2}\exp\{-(1/2)\kappa L_{jn}c_{jn}^2\}]. \tag{A.4}$$

Now write using Chebyshev inequality and Lemma 1,

$$A_2 \le C \sum_j \sum_m 2^{2j\nu}(2L_{jn})\Lambda^*_{jm} v_j n^{-1}$$

$$\times[\Lambda^*_{jm} I(2c_{jn}v_j n^{-1} > \Theta_{jm}) + E\{I(\Theta_{jm} - \hat{\Theta}_{jm} \ge \Theta_{jm}/2) I(\Theta_{jm} \ge 2c_{jn}v_j n^{-1})\}]$$

$$\le C \sum_j \sum_m 2^{2j\nu} L_{jn} \Lambda^*_{jm} v_j n^{-1}$$

$$\times[c_{jn} + L_{jn}^{-1} v_j n^{-1}(\Theta_{jm} + v_j n^{-1})\Theta_{jm}^{-2} I(\Theta_{jm} \ge 2c_{jn}v_j n^{-1})]$$

$$\le Cn^{-1} \sum_{j=j_0}^{J^*} \sum_{m=1}^{2^j/L_{jn}} 2^{2j\nu} L_{jn} \Lambda^*_{jm} v_j [c_{jn} + (L_{jn}c_{jn}^2)^{-1}]. \tag{A.5}$$

Combining the estimates for A_1 and A_2 we get

$$E\{\|\hat{f}_n^{(\nu)} - \hat{f}_n^{*(\nu)}\|^2\} \le Cn^{-1} \sum_{j=j_0}^{J^*} 2^{2j\nu} \sum_{m=1}^{2^j/L_{jn}} (2L_{jn}) \Lambda^*_{jm} v_j [(L_{jn}c_{jn}^2)^{-1} + c_{jn}]$$

$$+ Cn^{-1} \sum_{j=0}^{J^*} 2^{j(1+2\nu)} L_{jn}^{-1/2} v_j \exp(-\kappa L_{jn} c_{jn}^2/2). \tag{A.6}$$

Also a straightforward algebra shows that

$$E\{\|\hat{f}_n^{*(\nu)} - f^{(\nu)}\|^2\} \ge n^{-1} \sum_{j=j_0}^{J^*} 2^{2j\nu} \sum_{m=1}^{2^j/L_{jn}} (2L_{jn}) \Lambda^*_{jm} v_j. \tag{A.7}$$

Comparison of (A.6) with (A.7) yields the assertion of Theorem 1.

Now let us prove Lemma 1. We are proving only (A.3) because (A.2) is established similarly (and much simpler). Denote

$$\hat{\Theta} \stackrel{\text{def}}{=} (2L)^{-1} \sum_{k \in T} (\hat{\theta}_k^2 - n^{-1}\sigma_k^2) = (2L)^{-1} \sum_{k \in T} (\theta_k^2 + 2\theta_k n^{-1/2} \xi_k + n^{-1}(\xi_k^2 - \sigma_k^2))$$

$$\stackrel{\text{def}}{=} \Theta + 2n^{-1/2}(2L_{jn})^{-1} \sum_{k \in T} \theta_k \xi_k + n^{-1}(2L)^{-1} \sum_{k \in T} (\xi_k^2 - \sigma_k^2).$$

Here θ_k, $k \in T$ are wavelet coefficients from the block T, $\hat{\theta}_k$ are the corresponding empirical wavelet coefficients, $\xi_k = \hat{\theta}_k - \theta_k$ are zero mean errors and $\sigma_k^2 = \mathrm{Var}(\xi_k)$. Set $d = c_{jn}v_j$ and write,

$$P(\hat{\Theta} > dn^{-1},\ \Theta < dn^{-1}/2)$$

$$\le P(2n^{-1/2}\sum_{k\in T}\theta_k\xi_k + n^{-1}\sum_{k\in T}(\xi_k^2-\sigma_k^2) > Ldn^{-1},\ \Theta < dn^{-1}/2)$$

$$\le P(\sum_{k\in T}\theta_k\xi_k > (1/4)Ldn^{-1/2},\ \Theta < dn^{-1}/2) + P(\sum_{k\in T}(\xi_k^2-\sigma_k^2) > (1/2)Ld)$$

$$\stackrel{\text{def}}{=} B_1 + B_2. \tag{A.8}$$

Note that

$$\xi_k = \sum_{r=1}^n a_{kr}\epsilon_r, \quad \sum_{r=1}^n a_{kr}^2 = \mathrm{Var}(\xi_k) = \sigma_k^2, \quad \sum_{r=1}^n a_{jr}a_{ir} = \mathrm{cov}(\xi_j,\xi_i), \tag{A.9}$$

where the array $\{a_{kr}\}$ is defined by the prefilter–DBWT used and $\{\epsilon_r\}$ are the errors in (5.1).

Let us estimate B_1. In what follows $t > 0$. Write

$$B_1 = P\left(e^{t\sum_k\theta_k\xi_k} > e^{(1/4)tdLn^{-1/2}}\right) I(\Theta < dn^{-1}/2)$$

$$\le e^{-t(1/4)Ldn^{-1/2}} E\{e^{t\sum_k\theta_k\xi_k}\} I(\Theta < dn^{-1}/2). \tag{A.10}$$

Using (A.9) we can write

$$\sum_k \theta_k\xi_k = \sum_{r=1}^n \epsilon_r \sum_{k\in T}\theta_k a_{kr},$$

so now we can use the assumption that the errors $\epsilon_1,\dots,\epsilon_n$ are independent.

Using (5.4) we establish that for any u

$$E\{e^{u|\epsilon_r|}\} \le E\{e^{u^2/t_0} I(|\epsilon_r| \le u/t_0)\} + E\{e^{t_0\epsilon_r^2} I(|\epsilon_r| > u/t_0)\} \le Ce^{u^2/t_0}. \tag{A.11}$$

Then we continue (A.10) using the inequality (A.11),

$$B_1 \le e^{-t(1/4)Ldn^{-1/2}} \prod_{r=1}^n E\{e^{\epsilon_r t\sum_k\theta_k a_{kr}}\} I(\Theta < dn^{-1}/2)$$

$$\le Ce^{-t(1/4)Ldn^{-1/2}} \prod_{r=1}^n e^{(t^2/t_0)(\sum_k\theta_k a_{kr})^2} I(\Theta < dn^{-1}/2).$$

According to Assumption 1, for $\Theta < dn^{-1}/2$ the quadratic form

$$\sum_{r=1}^n \Big(\sum_{k\in T}\theta_k a_{kr}\Big)^2 = \sum_{j,i\in T}\theta_i\theta_j \sum_{r=1}^n a_{jr}a_{ir} = \sum_{j,i\in T}\theta_j\theta_i \mathrm{cov}(\xi_j,\xi_i)$$

is bounded from above,

$$\sum_{r=1}^{n}\Big(\sum_{k\in T}\theta_k a_{kr}\Big)^2 \le \lambda^* \sum_{k\in T}\theta_k^2 = \lambda^* 2L\Theta \le \lambda^* Ldn^{-1}. \tag{A.12}$$

Thus we conclude that

$$\begin{aligned} B_1 &\le C\exp(-t(1/4)Ldn^{-1/2} + t_0^{-1}t^2\lambda^* Ldn^{-1}) \\ &= C\exp(-Ld[t(1/4)n^{-1/2} - t_0^{-1}\lambda^* n^{-1}t^2]). \end{aligned}$$

If we set $t = (t_0/8\lambda^*)n^{1/2}$ then $[t(1/4)n^{-1/2} - t_0^{-1}\lambda^* n^{-1}t^2] = t_0/(64\lambda^*)$. This yields that

$$B_1 < C\exp(-(t_0/64\lambda^*)Ld). \tag{A.13}$$

Now we consider B_2. For the case of normal errors $\{\epsilon_l, l = 1, 2, \ldots, n\}$, the random quadratic form $\sum_{k\in T}(\xi_k^2 - \sigma_k^2)$ is equal to $\sum_{k\in T}(\eta_k^2 - \tilde\sigma_k^2)$, $\tilde\sigma_k^2 = E\{\eta_k^2\}$ where uncorrelated (and thus independent) normal $\{\eta_k, k \in T\}$ are obtained by an orthogonal transformation of $\{\xi_k, k \in T\}$. Note that Assumption 1 implies that $\tilde\sigma_k^2 \le \lambda^* \max_{k\in T}\sigma_k^2 \le \lambda^* v^*$.

Using Markov inequality we write for $t > 0$

$$B_2 = P\Big(\sum_{k\in T}(\xi_k^2 - \sigma_k^2) > (1/2)Ld\Big) = P\Big(\exp(t\sum_{k\in T}(\eta_k^2 - \tilde\sigma_k^2)) > \exp(tLd/2)\Big)$$

$$\le \exp(-tLd/2)E\{\exp(t\sum_{k\in T}(\eta_k^2 - \tilde\sigma_k^2))\} = \exp(-tLd/2)\prod_{k\in T}E\{\exp(t(\eta_k^2 - \tilde\sigma_k^2))\}.$$

For normal random η_k a direct calculation (or see proof of Lemma 12 in Petrov 1975, ch.3) shows that for sufficiently small t

$$\prod_{k\in T}E\{\exp(t(\eta_k^2 - \tilde\sigma_k^2))\} \le \exp(2L(\lambda^* v^*)^2 t^2).$$

Thus for some positive $\kappa_1 = \kappa_1(\lambda^*, v^*)$

$$B_2 \le \exp(-Ld[t/2 - 2d^{-1}(\lambda^* v^*)^2 t^2]) \le \exp(-Ld^2\kappa_1).$$

This together with (A.13) yields (A.3) for the case of normal errors.

Now consider the second type of errors defined in Assumption 2. Because ξ_k are m–dependent along each stream, we can split these errors into m groups T_l, $l = 1, 2, \ldots, m$ where within each subgroup the errors are independent. Write

$$B_2 \le \sum_{l=1}^{m} P\Big(\sum_{k\in T_l}(\xi_k^2 - \sigma_k^2) > (1/2m)Ld\Big).$$

Using Markov inequality we get

$$B_2 \le \sum_{l=1}^{m} e^{-tLd/(2m)} E\{\exp(t \sum_{k\in T_l} (\xi_k^2 - \sigma_k^2))\}$$

$$= \sum_{l=1}^{m} e^{-tLd/(2m)} \prod_{k\in T_l} E\{\exp(t(\xi_k^2 - \sigma_k^2))\}.$$

Note that $E\{\xi_k^2 - \sigma_k^2\} = 0$. This together with (5.4) and Lemma 12 in Petrov (1975, ch.3) imply that for all sufficiently small t the inequality $E\{\exp(t(\xi_k^2 - \sigma_k^2))\} \le \exp(Ct^2)$ holds. Thus for a positive κ_2

$$B_2 \le m \exp(-tLd/(2m) + CLt^2) \le C \exp(-\kappa_2 Ld^2).$$

This together with (A.13) implies (A.3). Lemma 1 is verified. Theorem 1 is proved.

References

Abramovich, F. and Silverman, B.W. (1998). Wavelet decomposition approaches to statistical inverse problems, *Biometrika*, **85**, 115–129.

Bui, T.D. and Chen, G. (1998). Translation–invariant denoising using multiwavelets, *Institute of Electrical and Electronics Engineers Transactions on Signal Processing*, **46**, 3414–3420.

Cai, T.T. (1999a). On adaptive wavelet estimation of a derivative and other related linear inverse problems. *Technical Report*, Dept. Statistics, Purdue Univ.

– – – – (1999b). Adaptive wavelet estimation: a block thresholding and oracle inequality approach. *The Annals of Statistics*, **27**, 898–924.

Daubechies, I. (1992). *Ten Lectures on Wavelets*, Philadelphia: SIAM.

Donoho, D.L. (1995). Nonlinear solution of linear inverse problems by wavelet–vaguelette decomposition, *Applied and Computational Harmonic Analysis*, **2**, 101–126.

Donovan, G., Geronimo, J., Hardin, D. and Massopust, P. (1996). Construction of orthogonal wavelets using fractal functions, *Journal of Mathematical Analysis*, **27**, 1158–1192.

Downie, T.R. and Silverman, B.M. (1998). The discrete multiple wavelet transform and thresholding methods, *Institute of Electrical and Electronics Engineer Transactions on Signal Processing*, **46**, 2558–2561.

Efromovich, S. and Pinsker, M. (1984). A learning algorithm for nonparametric filtering. *Automation and Remote Control*, **24**, 1434-1440.

Efromovich, S. (1998). Simultaneous sharp estimation of functions, *The Annals of Statistics*, **26**, 273–278.

Efromovich, S., Lakey, J., Pereyra, M.C. and Tymes, N. (1998). Data-driven and optimal denoising of a signal and recovery of its derivative using multiwavelets, *Technical Report*, Univ. of New Mexico.

Efromovich, S. (1999a). *Nonparametric Curve Estimation: Methods, Theory and Applications.* New York: Springer.

– – – – (1999b). Quasi–linear wavelet estimation, *Journal of the American Statistical Association*, **94**, 189–204.

Geronimo, J.S., Hardin, D.P. and Massopust, P.R. (1994). Fractal functions and wavelet expansions based on several scaling functions, *Journal of Approximation Theory*, **78**, pp. 373–401.

Hall, P, Kerkyacharian, G. and Picard D. (1998). Block threshold rules for curve estimation using kernel and wavelet methods, *The Annals of Statistics*, **26**, 922-942.

Hall, P., Kerkyacharian, G. and Picard, D. (1999). On the minimax optimality of block thresholded wavelet estimators, *Statistica Sinica*, **9**, 33-50.

Johnstone, I. and Silverman, B. (1997). Wavelet threshold estimators for data with correlated noise, *Journal of the Royal Statistical Society B*, **59**, 319–351.

Johnstone, I. (1999). Wavelet shrinkage for correlated data and inverse problems: adaptive results, *Statistica Sinica*, **9**, 51–83.

Kolaczyk, E.D. (1996). A wavelet shrinkage approach to tomographic image reconstruction, *Journal of the American Statistical Association*, **91**, 1079–1090.

Lakey, J., Massopust, P. and Pereyra, M. (1998). Divergence–free multiwavelets, *Approximation Theory* IX, C. Chui and L. Schumaker, eds., 161–168, Nashville: Vanderbilt U. Press.

Lemarié-Rieusset, P.G. (1992). Analyses multi-resolutions non orthogonales, commutation entre projecteuurs et derivation et ondelettes vecteurs a divergence nulle, *Revista de Matemáticas Aplicadas Iberoamericana*, **8**, 222-237.

Mallat, S. (1999). *A Wavelet Tour of Signal Processing,* New York: University Press.

Neumann, M.H. and Spokoiny, V. (1995). On the efficiency of wavelet estimators under arbitrary error distributions, *Mathematical Methods of Statistics*, **1**, 137–166.

Petrov, V.V. (1975). *Sums of Independent Random Variables.* London: Springer.

Strela, V. and Walden, A.T. (2001), Signal and image denoising via wavelet thresholding: orthogonal and biorthogonal, scalar and multiple wavelet transforms, *Nonlinear and Nonstationary Signal Processing*, Fitzgerald, W.J., Smith, R.L., Walden, A.T. and Young, P.C., eds. Cambridge: Cambridge University Press (to be published).

Strela, V., Heller, P.N., Strang, G., Topiwala, P., and Heil, C. (1999). The application of multiwavelet filter banks to signal and image processing, *Institute of Electrical and Electronics Engineers Transactions on Image Processing*, **8**, 548–563.

Xia, X.G., Geronimo, J., Hardin, D. and Suter, B. (1996). Design of prefilters for discrete multiwavelet transforms, *Institute of Electrical and Electronics Engineers Transactions on Signal Processing*, **44**, 25–35.

Vidakovic, B. (1999). *Statistical Modeling by Wavelets.* New York: Springer.

Sam Efromovich

Department of Mathematics and Statistics

The University of New Mexico

Albuquerque, NM 87131, USA

E-mail: efrom@math.unm.edu