# Data-Driven and Optimal Denoising of a Signal and Recovery of its Derivative Using Multiwavelets

Sam Efromovich, Joe Lakey, María Cristina Pereyra, and Nathaniel Tymes, Jr.

***Abstract*—Multiwavelets are relative newcomers into the world of wavelets. Thus, it has not been a surprise that the used methods of denoising are modified universal thresholding procedures developed for uniwavelets. On the other hand, the specific of a multiwavelet discrete transform is that typical errors are not identically distributed and correlated, whereas the theory of the universal thresholding is based on the assumption of identically distributed and independent normal errors. Thus, we suggest an alternative denoising procedure based on the Efromovich–Pinsker algorithm. We show that this procedure is optimal over a wide class of noise distributions. Moreover, together with a new cristina class of biorthogonal multiwavelets, which is introduced in this paper, the procedure implies an optimal method for recovering the derivative of a noisy signal. A Monte Carlo study supports these conclusions.**

***Index Terms*—Efromovich–Pinsker estimator, learning, nonparametric estimation.**

## I. Introduction

UNIWAVELETS are well known for their good approximation and data-compression properties. A discrete wavelet transform (DWT), which is an orthonormal transformation, makes uniwavelets very convenient for statistical applications. As an example, consider a classical denoising model $Y_l = f(l) + \sigma\epsilon_l$, $l = 1, 2, \ldots, n$, where $\{\epsilon_l\}$ is a white (uncorrelated) unit-variance noise, and the problem is to estimate (denoise, filter from the noise) the function $f$. If the noise is Gaussian, then DWT implies that empirical uniwavelet coefficients are contaminated by white Gaussian errors. This allows a practitioner to use a simple and reliable procedure of denoising—universal thresholding; see the discussion in [4], [8], [14], and [17].

Multiwavelets, due to a larger flexibility in constructing smooth, compactly supported and symmetric scaling functions, have even better approximation and data-compression properties; see the discussion in [2], [14], and [15]. However, a serious problem arises whenever multiwavelets are used for denoising. The problem is that a multiwavelet discrete transform is not orthogonal, and this implies correlated and not identically distributed errors in empirical wavelet coefficients [6], [15], [16]. Thus, the theory of uniwavelet threshold denoising, based on the assumption of identically distributed and independent normal errors, is no longer applicable. Several modifications of the classical universal thresholding have been suggested in [2], [6], [15], and [16]. Simulations for the classical case of white Gaussian noise indicate that these modified threshold procedures perform relatively well for small sample sizes, but so far, no dramatic improvement with respect to the uniwavelet denoising has been reported.

Thus, to make multiwavelets attractive for practitioners, different settings and estimators should be studied. Let us explain one of the possibilities.

Threshold methods are known by their bad performance for a non-Gaussian noise, in other words, they are not robust. Fig. 1 explains the situation. Here, the signal "blocks" supported by Matlab are observed in a Tukey noise with variance $\sigma^2$; see the two top diagrams. Recall that a Tukey random variable is created by a mixture of two zero-mean normal variables. In particular, here, $\epsilon = (u\zeta_1 + (1-u)4\zeta_2)/[P(u=1) + 16P(u=0)]^{1/2}$, where $\zeta_1$, $\zeta_2$, and $u$ are independent random variables, the first two are standard normal, and the last one is Bernoulli with the probability $P(u=1) = 0.9$. This is a classical example in the theory of robust estimation (see [8]).

The third diagram shows a signal recovered by the Symmlet 8 uniwavelet and the universal threshold procedure supported by Matlab. The fourth diagram shows how the classical GHM biwavelet, oversampled preprocessing, and the vector-threshold procedure of [15] perform (here and in what follows, the Strela software [15] is used). As we see, neither the threshold uniwavelet nor the vector-threshold biwavelet implies a reliable denoising because too many spikes created by the Tukey noise are unfiltered. The bottom denoised signal is obtained by using the biwavelet EP estimator discussed in this paper. We see that the Tukey noise is filtered almost perfectly and that the shape of the underlying signal is clearly exhibited.

Intensive Monte Carlo simulations support this conclusion. Define an experiment as a combination of sample size $n$, standard deviation $\sigma$, and the probability $P(u=1)$ in the Tukey model; note that Fig. 1 exhibits a particular experiment with $n = 1024$, $\sigma = 5$, and $P(u=1) = 0.9$. For every experiment, we have the following.

i) Five hundred independent Monte Carlo simulations were made.

ii) The sample means were calculated over these 500 simulations for each of the following ratios: the Integrated Squared Error (ISE) of the EP estimator/the ISE of the

Manuscript received October 25, 1999; revised September 26, 2003. The work of S. Efromovich was supported in part by the National Science Foundation under Grant DMS-0243606 and by the NSA under Grant MSPF-974-094. The work of J. Lakey, M. C. Pereyra, and N. Tymes, Jr. was supported by a Sandia Labs SURP grant. The associate editor coordinating the review of this paper and approving it for publication was Prof. Scott C. Douglas.

S. Efromovich and M. C. Pereyra are with the Department of Mathematics and Statistics, University of New Mexico, Albuquerque, NM 87131 USA (e-mail: efrom@math.unm.edu).

J. Lakey is with the Department of Mathematical Sciences, New Mexico State University, Las Cruces, NM 88003 USA.

N. Tymes, Jr. is with the Department of Accounting, Finance, Economics, and Statistics, Ferris State University, Big Rapids, MI 49307 USA.

Digital Object Identifier 10.1109/TSP.2003.822355

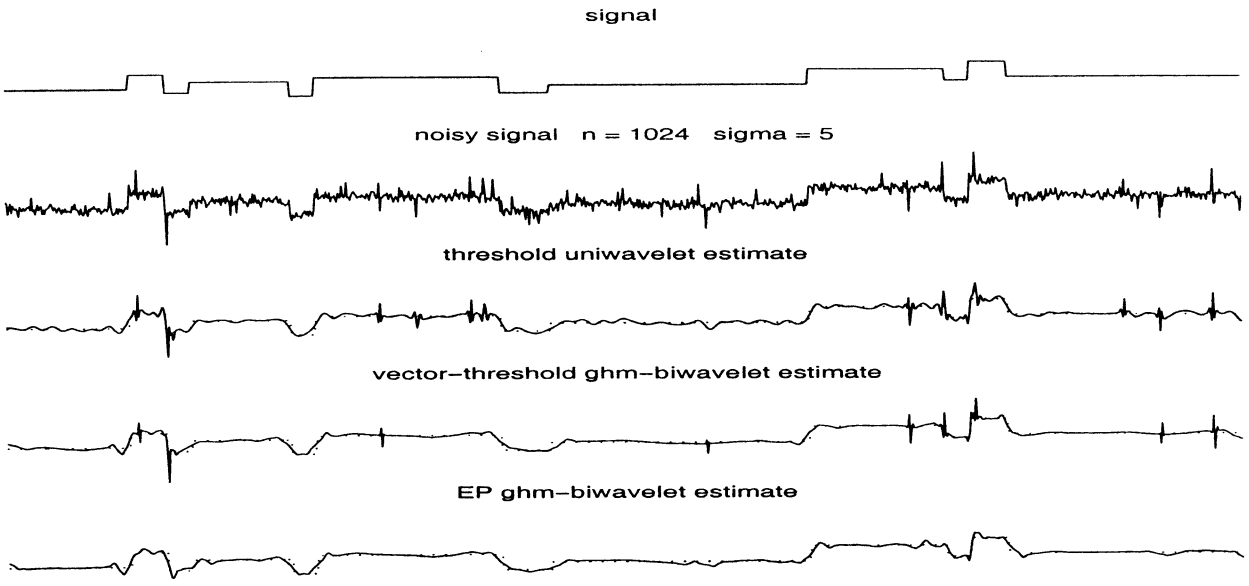


Fig. 1. Denoising the blocks signal from the Tukey noise by a universal threshold procedure for Symmlet 8 uniwavelet (the third diagram) by a vector-threshold procedure for the Geronimo–Hardin–Massopust (GHM) biwavelet (the fourth diagram), and by EP denoising for the GHM biwavelet (the bottom diagram). The solid and dotted lines show the estimates and the underlying signal, respectively.

uniwavelet estimator and the ISE of the EP estimator/the ISE of the vector-threshold multiwavelet estimator.

iii) The sample means (over these 500 simulations) were calculated for the following ratios: number of nonzero wavelet coefficients used by the EP estimator/number of nonzero wavelet coefficients used by the uniwavelet estimator and number of nonzero wavelet coefficients used by the EP estimator/number of nonzero wavelet coefficients used by the 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 I, which exhibits sample means of ratios of ISE and data compression between the EP estimates and the uniwavelet (or the vector-threshold) estimate. Each element in the body of the table is written as $A/B$, where $A$ is the sample mean of ratios for the EP estimator and the uniwavelet estimator, and $B$ is the sample mean of ratios for the EP estimator and the vector-threshold estimator. There is also an interesting byproduct of this table. The ratio $A/B$ 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 and data compression, the EP biwavelet estimator significantly outperforms the two threshold estimators. In addition, the biwavelet thresholding performs better than its uniwavelet counterpart. These numerical results justify the additional efforts involved in the using multiwavelets.

The primary goal of this paper, which is explored in Section II, is to explain the underlying idea of the EP estimator and its properties, in particular, its robustness over a wide class of noise distributions and the fact that this estimator, together with a special cristina biwavelet family introduced in Section III, allows us to suggest a new procedure for the recovery of the derivative of an underlying function.

TABLE I
RESULTS OF NUMERICAL STUDY

| n | 512 | | | 1024 | | | 2048 | | |
|---|---|---|---|---|---|---|---|---|---|
| $\sigma$ | 2 | 5 | 8 | 2 | 5 | 8 | 2 | 5 | 8 |
| $P(u=1)$ | ISE | | | | | | | | |
| 0.95 | 0.61/0.75 | 0.57/0.74 | 0.51/0.74 | 0.58/0.78 | 0.54/0.71 | 0.48/0.67 | 0.56/0.66 | 0.51/0.67 | 0.44/0.66 |
| 0.9 | 0.53/0.73 | 0.49/0.71 | 0.43/0.72 | 0.47/0.68 | 0.44/0.64 | 0.39/0.69 | 0.38/0.65 | 0.32/0.64 | 0.29/0.65 |
| 0.8 | 0.32/0.51 | 0.28/0.51 | 0.25/0.52 | 0.27/0.53 | 0.24/0.52 | 0.21/0.52 | 0.22/0.48 | 0.17/0.47 | 0.12/0.47 |
| $P(u=1)$ | Data Compression | | | | | | | | |
| 0.95 | 0.62/0.73 | 0.64/0.89 | 0.75/0.85 | 0.81/0.89 | 0.75/0.89 | 0.74/0.89 | 0.74/0.86 | 0.78/0.90 | 0.78/0.91 |
| 0.9 | 0.58/0.75 | 0.58/0.74 | 0.64/0.77 | 0.68/0.85 | 0.71/0.87 | 0.68/0.85 | 0.70/0.84 | 0.70/0.83 | 0.71/0.81 |
| 0.8 | 0.68/0.71 | 0.69/0.77 | 0.61/0.76 | 0.62/0.82 | 0.63/0.82 | 0.57/0.79 | 0.58/0.81 | 0.56/0.81 | 0.57/0.80 |

## II. MULTIWAVELET DENOISING

Several challenging problems need to be solved in order to make multiwavelet denoising attractive to practitioners.

First of all, a denoising procedure should not rely on the classical assumption about Gaussian noise; in other words, the procedure should be robust.

Second, using any discrete biwavelet transform (DBWT), which is defined by a prefilter and a biwavelet used, implies correlated errors in the empirical wavelet coefficients. Moreover, the correlation structure depends on the DBWT used. Thus, a denoising procedure should be optimal for possibly

correlated errors and, keeping in mind the wide variety of DBWT used, it is desirable that the denoising procedure is self-learning and does not rely on a specific DBWT. The latter makes multiwavelet denoising similar to uniwavelet denoising, which performs identically for all uniwavelets.

Finally, a procedure of denoising should be universal and allow a practitioner to solve a wide variety of related problems. In particular, we consider a problem of denoising a signal and the recovery of its derivative by employing the same estimator and the same family of wavelets. This feature is not available for uniwavelet threshold estimators where different bases and different threshold levels must be used for signal denoising and recovery of its derivative; see [1], [5], [14], and [17].

Let us describe a procedure that satisfies these three goals. We use the underlying idea of the Efromovich–Pinsker (EP) estimator that was first suggested for Fourier bases in [7]. The idea is to filter each wavelet coefficient using a weight (factor) that is defined by a block of wavelet coefficients. Namely, if $\theta_m$ is a wavelet coefficient and $\tilde{\theta}_m$ is a corresponding empirical wavelet coefficient obtained by a DBWT, then a corresponding EP estimator $\hat{\theta}_m$ of $\theta_m$ is defined as

$$\hat{\theta}_m = \hat{w}_m \tilde{\theta}_m \tag{1}$$

where the weight $\hat{w}_m$ is defined as

$$\hat{w}_m = \frac{\sum\limits_{i \in T_m} (\tilde{\theta}_i^2 - \mathrm{Var}(\tilde{\theta}_i))}{\sum\limits_{i \in T_m} \tilde{\theta}_i^2} I\left( \sum_{i \in T_m} [\tilde{\theta}_i^2 - c_m \mathrm{Var}(\tilde{\theta}_i)] \right). \tag{2}$$

Here, $T_m$ is a block of wavelet coefficients such that $m \in T_m$, $c_m$ is a constant, and $I(\cdot)$ is the indicator function.

Let us briefly explain why the EP procedure satisfies our goals (its asymptotic properties will be discussed later). The EP weight (2) mimics a corresponding EP oracle weight $w_m^* = \sum_{i \in T_m} \theta_i^2 / \sum_{i \in T_m} [\theta_i^2 + \mathrm{Var}(\tilde{\theta}_i)]$ (the reader can notice a similarity between the EP oracle and the famous Wiener filter; see [14]). It is clear that the weight (2) is a naive mimicking of the oracle $w_m^*$, and hard thresholding (the indicator function) is used because the oracle is non-negative.

As we see, no underlying distribution of errors is involved in the EP denoising technique, and this implies that the EP procedure is robust (more about this procedure can be found in [8, Sec. 7.4]). Moreover, EP filtering [see (1) and (2)] is based only on the variances of empirical wavelet coefficients. Thus, a broad class of errors in the empirical coefficients, including correlated ones and errors with infinite exponential moments, can be considered. In addition, a wide class of blocks and threshold levels implies optimal estimation, which entails a large flexibility.

Now, let us explain how a self-learning EP procedure can be constructed. The EP weight (2) depends on the sum of variances of empirical coefficients from a block. This sum may change dramatically when one DBWT is replaced by another one. On the other hand, because DBWT is a linear transformation, for a particular DBWT, this sum is identical for all zero-mean noise distributions with the same covariance structure. This remark is the key to understanding the learning part of the EP denoising procedure because it explores the sum of variances for a particular DBWT (or a particular software used) via Monte Carlo simulations where an appropriate noise is used as the input signal. In particular, because we are restricting our attention to white noise, in the learning part, white Gaussian noise can be used as the input signal.

Finally, EP filtering [see (1) and (2)] is universal whenever a block $T_k$ includes only wavelet coefficients from the same scale. Indeed, assume that the wavelet functions are differentiable; then, taking the derivative implies that all squared wavelet coefficients from the block and the corresponding variances are multiplied by the same factor. Then, it is easy to see that this factor is cancelled out in (2). Thus, no changes in EP filtering are required for recovery of the derivative. (Recall that no such property is known for the classical threshold estimators where different threshold levels are used for signal denoising and recovery of the derivative; see [1], [5], and [17].)

Of course, in general, the derivative of a wavelet function is no longer a wavelet function. This is a very serious complication because we can no longer use the same software for signal denoising and the recovery of its derivative; see [1] and [5]. Thus, we recommend a special cristina family of biwavelets where derivatives of wavelet functions are again elements of the family. This family is introduced in Section III.

We are now in a position to explain the particular EP denoising procedure used in numerical examples. 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. In addition, suppose that a signal $f(t)$ may be approximated by a biwavelet expansion

$$f_J(t) = \sum_{k=0}^{n/2^J-1} S_{J,k} \Phi_{J,k}(t) + \sum_{j=1}^{J} \sum_{k=0}^{n/2^j-1} D_{j,k} \Psi_{j,k}(t) \tag{3}$$

where $\Phi_{j,k}(t) = 2^{-j/2}\Phi(2^{-j}t - k)$, $\Psi_{j,k}(t) = 2^{-j/2}\Psi(2^{-j}t - k)$, and 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.

Similarly to uniwavelets, the default number of scales is $J = 6$. Numerical simulations revealed that this is a robust choice for sample sizes from several hundred to several thousand observations. Intensive Monte Carlo study was used to choose default values for the parameters of the EP estimator. For the scale functions and the two coarsest scales of wavelet functions, unit blocks and $c_m = 1$ are used. For the finer scales, the length of blocks increases. For the fourth scale, each block includes a single vector of coefficients, that is, an $m$th block includes the pair $(d_{4,m}^*, d_{4,m}^{**})$. For the third scale, a block includes two adjoint vectors, for the second scale it includes three adjoint vectors, and for the finest scale, it includes four adjoint vectors. In addition, for the finer scales, $c = 5$.

Now, let us explain how variances in (2) are estimated for the case $\sigma = 1$. We are following the above-outlined plan and use Monte Carlo simulations where standard white Gaussian noise is the input signal. Then, the sum (over the block) of squared empirical coefficients is an unbiased estimate of the estimated sum of variances over this block, and thus, a Monte Carlo approach can give us any desired accuracy of estimation of the sum of variances. The default number of repeated simulations

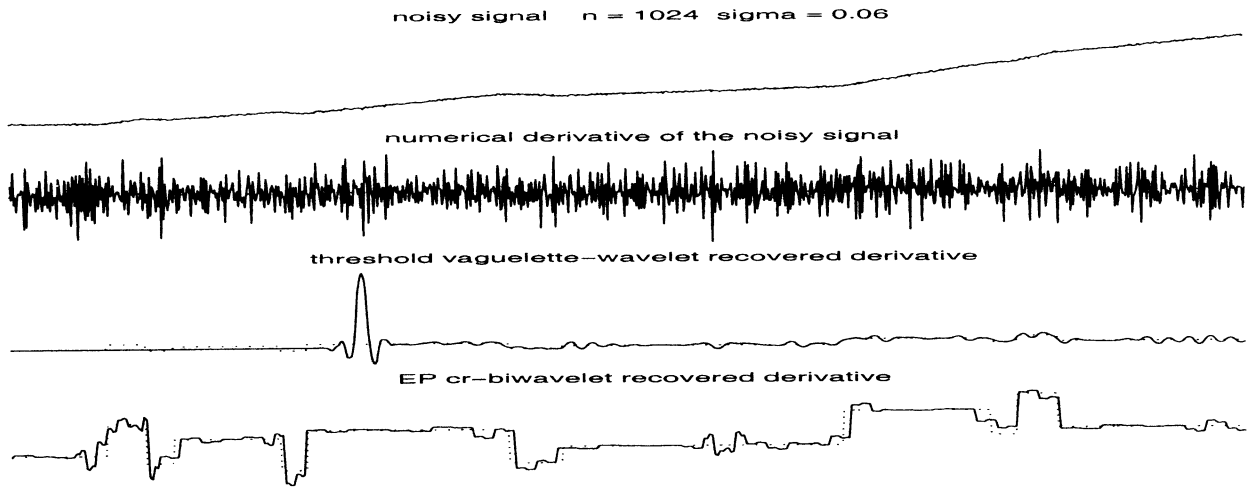


Fig. 2. Recovering derivative. The underlying derivative is the blocks signal. The solid and dotted lines show the estimates and the underlying signal, respectively.

is 10. This is the learning part of EP estimator that allows us to consider a DBWT (a software used) as a black box.

An unknown $\sigma^2$ is estimated using familiar robust procedures of variance estimation based on empirical wavelet coefficients of the finest scale. This is a standard procedure described in the literature (see [4], [8], [14], and [17]); then, an estimated $\hat{\sigma}$ is used as a scale factor. This describes the EP denoising procedure; note that it is completely data driven.

The suggested EP denoising implies an elementary procedure of recovery of the derivative whenever cristina biwavelets are used because the derivative $f_J^{(1)}$ of $f_J$ can be written as

$$f_J^{(1)}(t) = \sum_{k=0}^{n/2^J-1} 2^{-J} S_{J,k} T \Phi^{-}_{J,k}(t) - \sum_{j=1}^{J} \sum_{k=0}^{n/2^j-1} 2^{-j} D_{j,k} (-1) \Psi^{-}_{j,k}(t). \quad (4)$$

To shed light on (4), we note that the derivative of the wavelet vector-function $\Psi_{j,k}(t)$ is a minus $2^{-j}$ times the roughened wavelet vector-function $\Psi^{-}_{j,k}$ (similarly to trigonometric cosine bases, we get the factor minus one), and $T$ is a simple shift-type operator applied to the roughened scale vector-function $\Phi^{-}_{J,k}(t)$; see details in Section III. Thus, EP recovery of the derivative is based on the same EP filtered wavelet coefficients (1) that are rescaled according to (4), and then, the roughened cristina biwavelet basis is employed.

The two top diagrams in Fig. 2 illustrate the complexity of the problem. Note that here, the standard deviation of the white Gaussian noise is extremely small, and there is no problem in realizing the underlying signals. However, please look at the diagram of the numerical derivative of the noisy signal. The underlying derivative is the same "blocks" function as in Fig. 1, but it is impossible to recognize it. This is why the problem is called ill-posed because a small deviation in a signal can cause a huge deviation in its derivative.

We see that the threshold vaguelette-wavelet (Symmlet 8) estimator of [1] cannot recover the underlying derivative, whereas the EP estimator together with the cristina (cr) biwavelet does it. This recovery is a statistical "miracle." For larger noise, the recovery becomes unstable, and what we see in this figure shows us the limits of the EP estimation.

The uniwavelet estimator begins a stable recovery of the derivative for much smaller noise, where the boundary value of $\sigma$ is about 0.01. Such a case is shown in Fig. 3. Here again, the EP estimate is superior, but the uniwavelet estimate is comparable.

Repeated Monte Carlo simulations have revealed that the illustrated outcomes are typical. In particular, for $\sigma = 0.01$ and $n = 512$, 1024, and 2048, the sample means (based on 500 simulations) of ISE ratios between the EP biwavelet estimator and the threshold vaguelette-wavelet estimator are 0.22, 0.15, and 0.12, respectively. For $\sigma = 0.005$, the ratios are 0.33, 0.29, and 0.26, respectively. The corresponding data-compression ratios are 0.34, 0.36, and 0.37 and 0.29, 0.31, and 0.32.

Now, let us explain the main asymptotic property of the EP estimator. In what follows, it is assumed that biwavelet functions have finite supports. The underlying model is an equidistant nonparametric regression on [0,1], where observations are $Y_l = f(l/n) + \sigma\epsilon_l$, $l = 1, 2, \ldots$, and $\{\epsilon_l\}$ are zero-mean and unit-variance independent random variables (regression errors) with uniformly bounded eight moments. Note that neither normality nor identical distribution of the errors is required. We use an oracle approach that is traditional in the wavelet literature. Under mild assumptions, it is well known that the EP oracle

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

is an optimal estimator of $f$ and that its derivative is also an optimal estimate of $f^{(1)}$ over a large class of spatially inhomogeneous functions; see [3], [8], and [10]. Here, $2^{J^*-1} \le n^{1/(2N+1)} \le 2^{J^*}$, $N$ is the wavelet regularity, $T_{jmn} = \{k : \ L_{jn}(m-1) \le k < L_{jn}m\}$ are the blocks at the $j$th scale, and the blocks can depend on $n$, $L_{jn}$ is the length

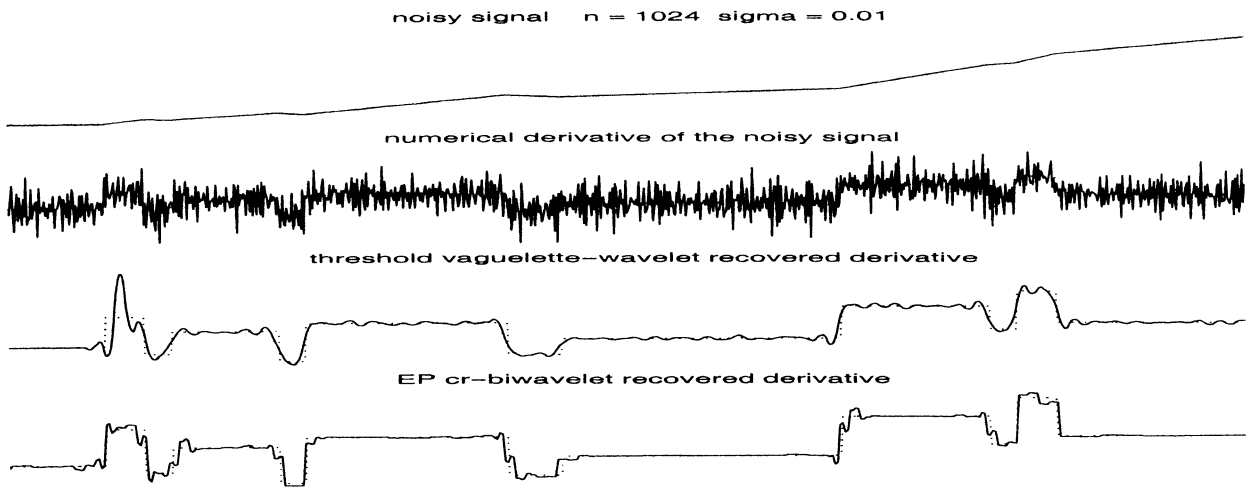


Fig. 3. Recovering derivative for a smaller noise than in Fig. 2.

of the blocks at the $j$th scale, $J^*$ is the rounded down $\log_2 n$, and $v_j = \sigma^{-2} L_{jn}^{-1} \sum_{k \in T_{jmn}} [\mathrm{Var}(\hat{d}^*_{j,k}) + \mathrm{Var}(\hat{d}^{**}_{j,k})]$. Note that $v_j = O(1/n)$.

It is not surprising that the EP oracle is a very good estimator because it knows the underlying function, but the oracle is the key element of the oracle approach, where a data-driven estimator is suggested that mimics the oracle and whose mean integrated squared error is of the same order as the mean integrated squared error of the oracle; see the discussion in [8] and [14].

It is not difficult to see that the following EP estimator is a naive mimicking of the oracle:

$$\hat{f}(t) = \sum_{k=0}^{2^{j_0}-1} \hat{S}_{j_0,k} \Phi_{j_0,k}(t) + \sum_{j=j_0}^{J^*} \sum_{m=1}^{2^j/L_{jn}} \frac{\sum_{k \in T_{jmn}} (\hat{D}_{j,k}\hat{D}^T_{j,k} - \hat{\sigma}^2 \hat{v}_j)}{\sum_{k \in T_{jmn}} \hat{D}_{j,k}\hat{D}^T_{j,k}} \times I\Big(L_{jn}^{-1} \sum_{k \in T_{jmn}} \hat{D}_{j,k}\hat{D}^T_{j,k} > c_{jn}\hat{\sigma}^2 \hat{v}_j\Big) \sum_{k \in T_{jmn}} \hat{D}_{j,k} \Psi_{j,k}(t). \quad (5)$$

Here, $\hat{\sigma}$ is a robust estimate of the scale factor $\sigma$, and $\hat{v}_j$ is a Monte Carlo estimate of $v_j$. Recall that the learning part of the EP estimator is used to estimate $v_j$ only because we do not want to use information about an underlying DBWT; otherwise, $v_j$ is defined by the underlying DBWT; see [15].

Let us assume that

$$\sum_{j=1}^{J^*} \Big(\frac{n}{2^j}\Big) L_{jn}^{-2} c_{jn}^{-3} < C \quad (6)$$

and

$$E\Big\{\Big(\sum_{k \in T_{jmn}} \Big[\hat{D}_{j,k}\hat{D}^T_{j,k} - \mathrm{Var}(\hat{d}^*_{j,k}) - \mathrm{Var}(\hat{d}^{**}_{j,k}) - D_{j,k}D^T_{j,k}\Big]\Big)^4\Big\} < Cn^{-2}L_{jn}^{-2}\Big(\sum_{k \in T_{jmn}} [D^T_{j,k}D_{j,k} + n^{-1}]\Big)^2. \quad (7)$$

*Theorem 1:* Supposing that biwavelet functions have finite supports, the assumption (6) and (7) holds, and an underlying function $f$ is not zero on [0,1]. Then, for $\nu = 0$ and some finite constant $C$, the following upper bound (oracle inequality) for mean integrated squared error of the EP estimate holds:

$$E\Big\{\int_0^1 (\hat{f}^{(\nu)}(x) - f^{(\nu)}(x))^2 dx\Big\} \le CE\Big\{\int_0^1 (\hat{f}^{*(\nu)}(x) - f^{(\nu)}(x))^2 dx\Big\}. \quad (8)$$

If, additionally, a cristina biwavelet is used and the underlying function $f$ is not constant on [0, 1], then (8) holds for $\nu = 0$, 1, that is, up to a constant, the EP estimator yields the same mean squared error of estimation of both $f$ and its derivative as the oracle.

*Outline of Proof:* The Parseval identity for orthonormal biwavelets and the fact that biorthogonal biwavelets form a Riesz basis imply $\int_0^1 (\hat{f}(x) - f(x))^2 dx \le C_1 \|\hat{f} - f\|^2$, $C_1 < \infty$, where $\|g\|^2$ denotes the sum of squared biwavelet coefficients of $g$. As a result, the problem is converted into the analysis of mean squared errors of the EP estimates of biwavelet coefficients. For the case of Fourier coefficients, this problem was considered in [7] and [8]. The new element here is that the errors in the empirical biwavelet coefficients are correlated. Nevertheless, it is possible to use the known proof for the case of biwavelets as well. Indeed, it is assumed that biwavelet functions have finite supports. This implies that along each scale, biwavelet coefficients are $m$-dependent (i.e., they are independent whenever the lag between coefficients is larger than $m$), where the $m$ depends only on the maximum length of support. This allows us to follow along the lines of the proof in [7] and establish (8).

It is necessary to note that verification of (7) for a general case may not be a simple issue. On the other hand, for a noise with the finite eight moment and for all known DBWT and biwavelets with finite supports, (7) is established straightforwardly. In particular, it holds for Tukey noise (recall that it has finite exponential moments) and for all DBWT supported by Strela software and the DBWT for cristina biwavelets. The interested reader can find the discussion and rigorous mathematical proofs in [9].

TABLE II
SCALING AND WAVELET COEFFICIENTS

| $j$ | scaling coefficient $C_j(s)$ | wavelet coefficient $D_j(s)$ |
|---|---|---|
| $-2$ | $\frac{1}{24}\begin{bmatrix} 0 & -(1+2s)\sqrt{2} \\ 0 & 0 \end{bmatrix}$ | $\frac{1}{24}\begin{bmatrix} 0 & -(1+2s)\sqrt{2} \\ 0 & -(2+4s) \end{bmatrix}$ |
| $-1$ | $\frac{1}{24}\begin{bmatrix} -2+8s & (5-2s)\sqrt{2} \\ 0 & 0 \end{bmatrix}$ | $\frac{1}{24}\begin{bmatrix} -2+8s & (5-2s)\sqrt{2} \\ \sqrt{2}(-2+8s) & 10-4s \end{bmatrix}$ |
| $0$ | $\frac{1}{24}\begin{bmatrix} 12 & (5-2s)\sqrt{2} \\ 0 & 8+4s \end{bmatrix}$ | $\frac{1}{24}\begin{bmatrix} -12 & (5-2s)\sqrt{2} \\ 0 & 4s-10 \end{bmatrix}$ |
| $1$ | $\frac{1}{24}\begin{bmatrix} -2+8s & -(1+2s)\sqrt{2} \\ 8\sqrt{2}(1-s) & 8+4s \end{bmatrix}$ | $\frac{1}{24}\begin{bmatrix} -2+8s & -(1+2s)\sqrt{2} \\ \sqrt{2}(2-8s) & 2+4s \end{bmatrix}$ |

## III. CRISTINA FAMILY OF BIORTHOGONAL BIWAVELETS

Using one software (that includes a wavelet and an estimator) for estimation of a function and another software (that includes another wavelet and another estimator) for recovery of its derivative is inconvenient, but so far, this has been the only way to solve this problem; see [1], [5], and [6]. Cristina biwavelets together with the EP estimator allow us to solve this problem using the same software and the same denoising procedure. This section is solely devoted to the cristina family of biwavelets.

We will use the notation $(1/2)\Phi(x/2) = \sum C_k \Phi(x-k)$ and $(1/2)\Psi(x/2) = \sum D_k \Phi(x-k)$ to denote the fundamental scaling and wavelet equations. The scaling vector is $\Phi(x) = (\phi^*(x), \phi^{**}(x))^T$, and the wavelet vector is $\Psi(x) = (\psi^*(x), \psi^{**}(x))^T$. The coefficients $C_k$ and $D_k$ are two-by-two matrices. We will consider a family of biwavelets introduced by Hardin and Morosovich (HM); see the discussion in [12]. When one imposes among the HM family the constraint that the scaling functions and wavelets should be symmetric, as we do here, the coefficients for the HM scaling and wavelet filters will depend on a parameter $s$. Furthermore, the parameter $\tilde{s}$ that is used to denote the biorthogonal MRA will be related to the parameter $s$ for the original MRA by $\tilde{s} = (1+2s)/(5s-2)$.

Continuity and approximation properties of the scaling/wavelet vectors depend on the values of $s$. The scaling filter, or transfer function, is a matrix with polynomial entries $H_s(z) = \sum C_k(s) z^k$, where the $C_k$ are the two-by-two scaling matrices with entries depending on the parameter $s$. The corresponding highpass or wavelet filter has the form $F_s(z) = \sum D_k(s) z^k$. Everything is set up so that the filters for the scaling and wavelet functions in the multiresolution analysis biorthogonal to that generated by $H_s(z)$ have the form $H_{\tilde{s}}(z)$ and $F_{\tilde{s}}(z)$, respectively. The coefficients are listed in Table II.

The functions are Lipschitz continuous of order $\alpha$ if $|s| < 2^{-\alpha}$ but can be discontinuous for $|s| \geq 1$. They become piecewise smooth as $s \to 2/5$, but their duals, while still having compact support, become distributions of increasingly higher order. The orthogonal cases correspond to $s = 1$ and $s = -1/5$. The functions are not continuous when $s = 1$, although they are piecewise linear in that case. In what follows, we will always assume that $|s| \leq 1/2$ in the examples $s = -0.3$.

Lemarié [13] showed that given a pair of biorthogonal scalar MRAs—that is, ones generated by a single scaling function as opposed to multiple ones—with sufficient regularity, one could build from these another pair such that the derivatives of the new wavelets are essentially the original wavelets. Strela (see the discussion in [15]) found a technique that amounts to the multiwavelet version of Lemarié's technique. In the scalar case, the trick is to add a zero to the dual lowpass filter at $z = -1$. This is equivalent to multiplying the dual lowpass filter by $(1+z)/2$. In the multiwavelet setup, the lowpass filter is vector-valued, and the analogous procedure amounts to adding an eigenvalue zero to the lowpass filter at $z = -1$ in an appropriate manner. As is typical of multiwavelets, there is some flexibility in carrying out this extension of Lemarié's method. The method is encoded in Strela's two-scale transform. It amounts to finding a suitable transition matrix $M$ and then forming the new lowpass filter $H_+(z) = (1/2)M^*(z^2)\tilde{H}(z)M^{*-1}(z)$. Therefore, $M$ accomplishes a transition between a pair of two-scale MRAs. In the scalar MRA case, the role of $M^*$ is played by $1-z$; the property desired of $M^*$ is that it has a simple eigenvalue zero at $z = 1$ whose eigenvector is shared with the unit eigenvalue of $\tilde{H}(1)$. Existence of the latter is guaranteed by the scaling property of $\tilde{H}$. Further issues in the design of $M$ include preservation of symmetries and compact support. These issues are discussed in [11] and [15]. For example, the fact that the smoothing and roughening operators in Table III are related by taking adjoints depends on symmetry. Those considerations led to the particular choice of $M$ that we use below. Actually, our particular normalization of the HM wavelets allows us to use a single matrix $M$ for any value of the HM parameter $s$. Table III summarizes the procedure for forming the new families of dual MRAs. It also records the relationships between the smoothed and roughened filters.

TABLE III
SMOOTHENED AND ROUGHENED MRAS

| | | |
|---|---|---|
| smoothened scaling | $H_+(z) = \frac{1}{2}M^*(z^2)\tilde{H}(z)M^{*-1}(z)$ | $\frac{d}{dx}\Phi_+(x) = -T_{M^*}\Phi(x)$ |
| roughened scaling | $H_-(z) = 2M^{-1}(z^2)H(z)M(z)$ | $\frac{d}{dx}\Phi(x) = T_M\Phi_-(x)$ |
| smoothened wavelet | $F_+(z) = \frac{1}{2}\tilde{F}(z)M^{*-1}(z)$ | $\frac{d}{dx}\Psi_+(x) = -\Psi(x)$ |
| roughened wavelet | $F_-(z) = 2F(z)M(z)$ | $\frac{d}{dx}\Psi(x) = \Psi_-(x)$ |

TABLE IV
SMOOTHED AND ROUGHENED SCALING AND WAVELET COEFFICIENTS

| smooth scaling $C_k^+(s)$ | smooth wavelet $D_k^+(s)$ | rough scaling $C_k^-(s)$ | rough wavelet $D_k^-(s)$ |
|---|---|---|---|
| $\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$ | $\frac{1}{12}\begin{bmatrix} -\frac{1}{2}-s & \frac{1}{2}+s \\ -\frac{1}{2}-s & \frac{1}{2}+s \end{bmatrix}$ | $\frac{1+2s}{12}\begin{bmatrix} -\sqrt{2} & \sqrt{2} \\ -2 & 2 \end{bmatrix}$ |
| $\begin{bmatrix} \frac{1}{4} & \frac{(1-s)}{4(5s-2)} \\ \frac{3}{8} & \frac{(4-s)}{8(5s-2)} \end{bmatrix}$ | $\begin{bmatrix} \frac{\sqrt{2}}{32} & \frac{\sqrt{2}}{32}\frac{2+s}{5s-2} \\ \frac{1}{16} & \frac{1}{16}\frac{s+2}{(5s-2)} \end{bmatrix}$ | $\frac{1}{12}\begin{bmatrix} 3 & -4+10s \\ 3 & -4+10s \end{bmatrix}$ | $\frac{1}{12}\begin{bmatrix} 6\sqrt{2} & \sqrt{2}(20s-8) \\ 12 & 40s-16 \end{bmatrix}$ |
| $\begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{4} \end{bmatrix}$ | $\begin{bmatrix} 0 & -\frac{1}{16}\sqrt{2} \\ -\frac{1}{8} & 0 \end{bmatrix}$ | $\frac{1}{12}\begin{bmatrix} 7+2s & 0 \\ 0 & 7+2s \end{bmatrix}$ | $\frac{1}{12}\begin{bmatrix} 0 & \sqrt{2}(4s-34) \\ 8s-20 & 0 \end{bmatrix}$ |
| $\begin{bmatrix} \frac{1}{4} & \frac{(s-1)}{4(5s-2)} \\ -\frac{3}{8} & \frac{(4-s)}{8(5s-2)} \end{bmatrix}$ | $\begin{bmatrix} -\frac{\sqrt{2}}{32} & \frac{\sqrt{2}}{32}\frac{s+2}{5s-2} \\ \frac{1}{16} & -\frac{1}{16}\frac{s+2}{(5s-2)} \end{bmatrix}$ | $\frac{1}{12}\begin{bmatrix} 3 & 4-10s \\ -3 & 10s-4 \end{bmatrix}$ | $\frac{1}{12}\begin{bmatrix} -6\sqrt{2} & \sqrt{2}(20s-8) \\ 12 & 16-40s \end{bmatrix}$ |
| $\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$ | $\frac{1}{12}\begin{bmatrix} -\frac{1}{2}-s & -\frac{1}{2}-s \\ \frac{1}{2}+s & \frac{1}{2}+s \end{bmatrix}$ | $\frac{(1+2s)}{12}\begin{bmatrix} \sqrt{2} & \sqrt{2} \\ -2 & -2 \end{bmatrix}$ |

It is a simple matter to check that the conditions of biorthogonality are preserved. The other conditions are similarly verified. The operator $T_{M^*}$ is an operator-valued matrix whose entries are polynomials in the shift operator. Its symbol is $M^*$. Therefore, at the level of wavelets, this process really amounts to integrating or differentiating the wavelet components.

The matrix $H_s(1) = (1/3)\begin{bmatrix} 1+2s & \sqrt{2}(1-s) \\ \sqrt{2}(1-s) & 2+s \end{bmatrix}$ has eigenvalues $s$ and 1. For $\lambda = 1$, the corresponding eigenvector is $[\alpha \quad \beta]^T = \alpha[1 \quad \sqrt{2}]^T$ independent of $s$. One easily checks that for the matrix $M^*(z) = \begin{bmatrix} 0 & 1-(1/z) \\ 2\sqrt{2} & -(1/z)-1 \end{bmatrix}$, $[1 \quad \sqrt{2}]^T$ generates the kernel for $M^*(1)$; therefore, Strela's eigenvector criterion is satisfied. Up to multiplication by $z^k$, this is the unique transition matrix that preserves symmetry and increases the support of the HM scaling vector—that is, the sum of the supports of its components—by one.

Let us emphasize that the smoothing transition is applied to the $\tilde{s}$ filter while the roughening transition is applied to the $s$ filter. Then, the recipe for smoothing and roughening the filters yields the corresponding coefficients for the smoothened and roughed scaling and multiwavelets. We show this in Table IV (here, $k$ is running from $k = -2$ in the top row to $k = 2$ in the bottom row). Finally, note that the first components of the smoothed and roughened filters are symmetric about $x = 0$, whereas the second components are both antisymmetric about $x = 0$.

## REFERENCES


[1] F. Abramovich and B. W. Silverman, "Wavelet decomposition approaches to statistical inverse problems," *Biometrica*, vol. 85, pp. 115–129, 1998.

[2] T. D. Bui and G. Chen, "Translation-invariant denoising using multiwavelets," *IEEE Trans. Signal Processing*, vol. 46, pp. 3414–3420, Dec. 1998.

[3] T. T. Cai, "Adaptive wavelet estimation: A block thresholding and oracle inequality approach," *Ann. Statist.*, vol. 27, pp. 898–924, 1999.

[4] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrica*, vol. 81, pp. 425–455, 1994.

[5] D. L. Donoho, "Nonlinear solution of linear inverse problems by wavelet-vaguelette decomposition," *Appl. Comput. Harm. Anal.*, vol. 2, pp. 101–126, 1995.

[6] T. R. Downie and B. W. Silverman, "The discrete multiple wavelet transform and thresholding methods," *IEEE Trans. Signal Processing*, vol. 46, pp. 2558–2561, Sept. 1998.

[7] S. Efromovich and M. Pinsker, "A learning algorithm for nonparametric filtering," *Automat. Remote Contr.*, vol. 11, pp. 1434–1440, 1984.
[8] S. Efromovich, *Nonparametric Curve Estimation: Methods, Theory and Applications*. New York: Springer, 1999.
[9] ——, "Multiwavelets and signal denoising," *Sankhya*, ser. A, vol. 63, pp. 369–393, 2001.
[10] P. Hall, G. Kerkyacharian, and D. Picard, "Block threshold rules for curve estimation using kernel and wavelet methods," *Ann. Statist.*, vol. 26, pp. 922–942, 1998.
[11] J. Lakey, P. Massopust, and M. Pereyra, "Divergence-free multiwavelets," in *Approximation Theory*, C. Chui and L. Schumaker, Eds. Nashville, TN: Vanderbilt Univ. Press, 1998, vol. IX, pp. 161–168.
[12] J. Lakey and C. Pereyra, "Divergence-free multiwavelets on rectangular domains," *Lecture Notes Pure Applied Math.*, vol. 212, pp. 203–240, 2000.
[13] P. G. Lemarié-Rieusset, "Analyzes multi-resolutions non orthogonales, commutation entre projecteuurs et derivation et ondelettes vecteurs a divergence nulle," *Rev. Mat. Iberoamericana*, vol. 8, pp. 222–237, 1992.
[14] S. Mallat, *A Wavelet Tour of Signal Processing*. New York: Cambridge Univ. Press, 1999.
[15] V. Strela and A. T. Walden, "Signal and image denoising via wavelet thresholding: Orthogonal and biorthogonal, scalar and multiple wavelet transforms," in *Nonlinear and Nonstationary Signal Processing*, W. J. Fitzgerald, R. L. Smith, A. T. Walden, and P. C. Young, Eds. Cambridge, U.K.: Cambridge Univ. Press, 2001, pp. 124–157.
[16] V. Strela, P. N. Heller, G. Strang, P. Topiwala, and C. Heil, "The application of multiwavelet filter banks to image processing," *IEEE Trans. Image Processing*, vol. 8, pp. 548–563, Apr. 1999.
[17] B. Vidakovic, *Statistical Modeling by Wavelets*. New York: Springer, 1999.

**Sam Efromovich** received the M.S. and Ph.D. degrees in information theory and statistics from Moscow Physico-Technical Institute, Moscow, U.S.S.R., in 1974 and 1978, respectively, and the Doctoral degree in control theory and applied statistics from the Moscow Institute of Steel in 1986.

Since 1991, he has been a Professor of mathematics and statistics, University of New Mexico, Albuquerque. His research interests include information theory, control theory, nonparametric curve estimation, and mathematical and applied statistics.

Dr. Efromovich is a Fellow of IMS.

**Joe Lakey** was born in 1963. He received the Ph.D. degree in mathematics from the University of Maryland, College Park, in 1991.

Hed has held visiting positions at The University of Texas, Austin, and Texas A&M University, College Station. Currently, he is an Associate Professor of mathematics at New Mexico State University, Las Cruces. His research interests include pure and applied harmonic analysis and complex systems.

**María Cristina Pereyra** was born in 1964 in Buenos Aires, Argentina. She received the B.S. degree in mathematics from the Universidad Central de Venezuela, Caracas, in 1987 and the M.S. and Ph.D. degrees in mathematics from Yale University, New Haven, CT, in 1989 and 1993, respectively. Her thesis advisor was P. W. Jones of the Mathematics Department, Princeton University, Princeton, NJ.

She is currently a Professor with the Department of Mathematics and Statistics, the University of New Mexico, Albuquerque. Her research interests include pure and applied harmonic analysis, in particular, wavelets and their applications.

Dr. Pereyra is a Member of the Institute for Advanced Studies at Princeton University.

**Nathaniel Tymes, Jr.** was born February 10, 1959, in Montgomery, AL. He received the B.S. degree in mathematics from the University of Alabama, Tuscaloosa, in 1981, the M.Div degree from United Theological Seminary, Dayton, OH, in 1985, the M.S. degree in operations research from the Air Force Institute of Technology, Wright–Patterson AFB, OH, in 1987, and the Ph.D. degree in statistics from the University of New Mexico in 2002.

He is an Assistant Professor of statistics with the College of Business, Ferris State University, Big Rapids, MI.

Dr. Tymes was selected to appear in the 2004 edition of *Who's Who in America*.