EUSIPCO 2013 1569743427

NOISE REDUCTION IN AN ADAPTIVE MULTISCALE LOCAL POLYNOMIAL DECOMPOSITION

Maarten Jansen *

Université Libre de Bruxelles
Depts. of Mathematics & Computer Science
B-1050 Brussels

ABSTRACT

This paper investigates the selection of coefficients in an adaptive multiscale local polynomial decomposition. The multiscale local polynomial (MLP) decomposition is a slightly overcomplete alternative for a critically downsampled fast wavelet transform. Thanks to the redundancy, the MLP transform reconciles numerically well conditioned analyses and syntheses with smooth reconstructions for data observed on irregular point sets on a real line. The MLP can also be seen as an extension of a Burt and Adelson's Laplacian pyramid on irregular point sets, or as a new lifting scheme where the classical interpolating prediction is replaced by smoothing prediction, using statistical nonparametric estimation techniques. The MLP allows easy extension towards adaptive decompositions, but the adaptivity is incompatible with some of the design options for good numerical condition. This paper implements an adaptive decomposition based on techniques from statistical testing and investigates noise reduction within the adaptive scheme with special attention to the numerical condition of the decomposition.

Index Terms— Wavelets, local polynomial, kernel smoothing, lifting, adaptive

1. INTRODUCTION

The multiscale local polynomial (MLP) transform has been introduced [1] as a slightly overcomplete lifting scheme, using statistical nonparametric estimation techniques as basic building block. These nonparametric estimation techniques include spline, kernel and local polynomial [2] smoothing methods. They replace interpolating prediction as basic principle in the definition of wavelet or detail coefficients as local offsets between an observation and a prediction based on the adjacent observations. One of the objectives of using smoothing instead of interpolation is to combine the smooth reconstructions from a linear nonparametric estimation method and the powerful nonlinear wavelet coefficient selection methods, such as thresholding. The nonlinear approach is necessary for signals with jumps, while the use of linear smoothing between the jumps aims at smoother and sparser representations of the the intervals between the jumps, with less false discoveries of large detail coefficients. A second objective is a straightforward application of the method on data observed on nonequidistant points. Wavelet decompositions on nonequidistant point sets [3, 4, 5, 6, 7] have long seen problems in combining a fast decomposition, good numerical condition and smooth reconstruction. For both numerical condition and smooth reconstruction, smoothing prediction has the potential to outperform interpolation as basic tool. It is less sensitive to local errors, and obviously produces smooth predictions, up to degree that is easily controlled by the user. Nevertheless, for reasons of continuity, smoothing requires a slight redundancy in the decomposition [1]. Critically downsampled multiscale transforms on irregular structures can be numerically well conditioned, even orthogonal but only so in a context where smoothness does not play a role. This is the case for graphical data [8]. The combination of smoothness, numerical condition and critical downsampling seems to be impossible on inequidistant data. The overcomplete decomposition can be seen as a lifted and nonequispaced version of a Laplacian pyramid [9]. Laplacian pyramids have been analyzed, among others, in a context of lifting [10, 11] and frame theory [12], for data on equispaced grids and images. It is interesting to remark that the MLP, although developed for nonequispaced data in the first place, performs surprisingly well in image denoising, even compared to the non-decimated wavelet transform using the Cohen-Daubechies-Feauveau wavelets with less dissimilar lengths [13]. This family of wavelets is popular in image processing, and using a nondecimated version generally improves the noise reduction performance. It should be noted that the non-decimated wavelet transform has more redundancy than the MLP, and yet the latter performs slightly better.

Another benefit from the multiscale local polynomial decomposition, compared to classical wavelets, is that the successive resolution levels do not need to represent dyadic scales. Indeed, thanks to the slight redundancy, the subsam-

*Research support by the IAP research network grant nr. P7/06 of theBelgian government (Belgian Science Policy) is gratefully acknowledged.

pling rate can be controlled by the user, together with the bandwidth of the smoothing operation within each scale.

This paper concentrates on data-adaptive versions of multiscale local polynomial transforms. Data-adaptivity is facilitated by the flexibility in bandwidths, subdivision rate and adopted smoothing routine with the slightly redundant scheme. Among the large variety of adaptive realizations, the focus in this paper is on an implementation that uses truncated kernels and statistical testing in order to capture the precise location of possible jumps or singularities during the decomposition phase. Jump-adaptive decompositions have the advantage that reconstruction from processed coefficients does not suffer from missed discoveries (false negatives), leading to sharper reconstruction near the edges.

2. MULTISCALE LOCAL POLYNOMIALS

The multiscale local polynomial decomposition, depicted in Figure 1, is constructed as a Laplacian pyramid extended using lifting operations. Let $\boldsymbol{t}_{j+1}$ be the grid of sample locations at fine scale $j+1$ and $\boldsymbol{s}_{j+1}$ the corresponding signal values. Detail coefficients, also named offsets or wavelet coefficients are the difference between $\boldsymbol{s}_{j+1}$ and a prediction based on a subsampled lowpass version of the signal $\mathrm{P_j} \cdot (\uparrow 2)(\downarrow 2)\widetilde{\mathrm{H}}_j \cdot \boldsymbol{s}_{j+1}$. Unlike in critically downsampled wavelet transforms, subsampling or decimation does not take place on the detail coefficients. The reason for this, further elaborated in [13], is that the prediction filter P_j takes the form of a non-interpolating smoothing operation. More precisely, suppose the smoothing operation is given by $P_j(u; \boldsymbol{t}_j, \boldsymbol{s}_j)$, meaning that locations $\boldsymbol{t}_j$ and values $\boldsymbol{s}_j$ are used to contruct a smoothing curve for evaluation in u. Then the corresponding filter P_j is a matrix with elements $\mathrm{P}_{j,k,\ell} = P_j(t_{j+1,k}; \boldsymbol{t}_j, \delta[\boldsymbol{t}_j - t_{j,\ell}])$, with $\delta[t]$ the Kronecker delta impulse for which $\delta[0] = 1$ and $\delta[t] = 0$ if $t \neq 0$. Interpolating prediction, common in lifting schemes, has the continuity property that

$$\lim_{u \to t_{j,k}} P_j(u; \boldsymbol{t}_j, \boldsymbol{s}_j) = s_{j,k}. \tag{1}$$

Expression (1) is a necessary condition for continuous and smooth basis or frame functions.

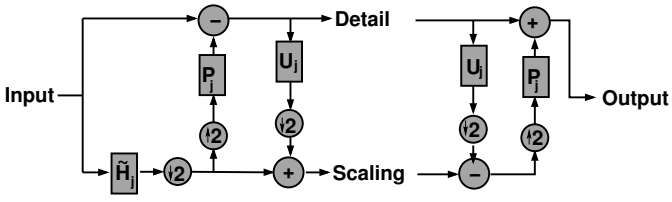

Fig. 1. One interscale step in a forward and inverse multiscale local polynomial transform. The inverse transform is not unique, because of the redundancy in the decomposition.

The resulting multiscale transform thus includes the following steps:

- **Copy** the input into two branches.
- **Pre-filter** and **Subsample** (downsample, decimate) the input on one of the branches, i.e., $\boldsymbol{s}_j^{[0]} = (\downarrow 2)\widetilde{\mathrm{H}}_j \cdot \boldsymbol{s}_{j+1}$, as well as the corresponding grid of locations, $\boldsymbol{t}_j = (\downarrow 2)\boldsymbol{t}_{j+1}$. The prefilter has a smoothing or any other lowpass effect. Obviously, downsampling and upsampling can always be incorporated into the filter before or right after the operation. Downsampling and upsampling symbols in the scheme merely emphasize the change of vector size.
- **Predict** the other branch by another smoothing operation on the subsampled branch. The offset between the original values and the prediction is stored as detail- or wavelet coefficients, i.e.,

$$\boldsymbol{w}_j = \boldsymbol{s}_{j+1} - \mathrm{P}_j \cdot (\uparrow 2)\boldsymbol{s}_j^{[0]}. \tag{2}$$

- **Update** the subsampled branch to get

$$\boldsymbol{s}_j = \boldsymbol{s}_j^{[0]} + (\downarrow 2)\mathrm{U}_j \cdot \boldsymbol{w}_j. \tag{3}$$

The multiscale iteration is initialized by $s_{J,k} = Y_k$ and $t_{J,k} = t_k$, where $\boldsymbol{t}$ is the observational grid and $\boldsymbol{Y}$ the observed signal.

3. DESIGN OF THE NONSTATIONARY FILTERS

3.1. The update filter

The coarse scaling coefficients have undergone three filter operations in the transition from fine to coarse. The first, prefilter, step is primarily an anti-aliasing operation. It contributes to numerical condition, which can further be improved by the third, update filter. The update should be designed such that the reconstruction from a detail offset carries no DC-component. This so-called primal vanishing moment condition is necessary for a numerically well conditioned decomposition of signals with jumps [14]. Just as in classical wavelet theory, the design of the update operations U_j for primal vanishing moments requires the introduction of scaling basis functions that can be associated with the refinement or prediction scheme P_j. More precisely, let $\Phi_j(t)$ denote a row vector of functions such that any expansion $\Phi_j(t)\boldsymbol{s}_j$ can be refined or subdivided as $\Phi_j(t)\boldsymbol{s}_j = \Phi_{j+1}(t)\boldsymbol{s}_{j+1}$ with zero detail coefficients $\boldsymbol{w}_j = \boldsymbol{0}$. By checking which $\boldsymbol{s}_{j+1}$ follows from refinement of the identity matrix, i.e., for $\boldsymbol{s}_j = I_j$, we can find expressions for the basis functions $\Phi_j(t)$. The refinement is an instance of the inverse transform, $\boldsymbol{s}_{j+1} = \mathrm{P_j}(\uparrow 2)\boldsymbol{s}_j$, depicted in Figure 1. The basis functions thus follow recursively as $\Phi_j(t) = \Phi_{j+1}(t) \cdot \mathrm{P}_j(\uparrow 2)$. The column vectors of their moments are $M_j^{(p)} = \int_{-\infty}^{\infty} \Phi_j^T(t) t^p dt$. The primal vanishing moment condition can then be expressed as a set of equations in U_j [1]

$$M_{j+1}^{(p)} = \mathrm{U}_j^T \cdot M_j^{(p)}, \tag{4}$$

for $p = 0, \ldots, n-1$. The higher the number of primal vanishing moments n, the more entries in the matrix U_j are nonzero. For $n = 2$, the primal vanishing moment condition can be fulfilled with a two-taps filter in U_j. More precisely [7], for $l \in \{0, 1\}$,

$$U_{j,k,k+l} = (-1)^l \frac{M^{(0)}_{j+1,2k+1}}{M^{(0)}_{j,k+l}} \cdot \frac{\overline{t}_{j,k+1-l} - \overline{t}_{j+1,2k+1}}{\overline{t}_{j,k+1} - \overline{t}_{j,k}},$$

where

$$\overline{t}_{j,k} = \frac{M^{(1)}_{j,k}}{M^{(0)}_{j,k}} = \frac{\int_{-\infty}^{\infty} t\varphi_{j,k}(t)\,dt}{\int_{-\infty}^{\infty} \varphi_{j,k}(t)\,dt},$$

which can be seen as a mean location of scaling function $\varphi_{j,k}(t)$. Note, however, that $\overline{t}_{j,k}$ may fall outside the support of $\varphi_{j,k}(t)$ if $\varphi_{j,k}(t) < 0$ for some values $t \in \mathbb{R}$. In order to bound the update coefficients $U_{j,k,k+l}$ and hence keep the transformation numerically well conditioned, it is necessary that the fine scaling function is "located" between the two coarse scaling functions, i.e.,

$$\overline{t}_{j,k} \leq \overline{t}_{j+1,2k+1} \leq \overline{t}_{j,k+1}. \tag{5}$$

The basis functions can be reordered to impose this condition. This reordering may not correspond to the "natural" ordering of the basis functions, i.e., the locations of the supports of the functions. In that case, a wavelet coefficient may refer to multiply located basis functions. Operations on the wavelet coefficients (such as thresholding) may have an unwanted non-local side effect. The combination of numerical condition, smoothness of reconstruction and locality of operations is thus not trivial for nonequidistant samples. Uncareful definition of the update step, proposed for controlling the numerical condition of the transform for signals with jumps, may introduce new numerical problems.

3.2. The prediction operation

Numerical condition motivates not only the use of an update step, it is also promoted by imposing the prediction operation in the second filter to be convex. Convexity is realized by defining the kth component of the prediction as $[\mathrm{P}_j(\uparrow 2)\boldsymbol{s}_j]_k = P_j(t_{j+1,2k+1}; \boldsymbol{t}_{j+1,e})$, where $P_j(t; \boldsymbol{t}_{j+1,e})$ is a smoothing function that in each point t is constructed as a local polynomial of degree $\tilde{n}$. This can be formalized as $P_j(t; \boldsymbol{t}_{j+1,e}) = \mathrm{T}^{(\tilde{n})}(t)\widehat{\boldsymbol{\beta}}$, where $\widehat{\boldsymbol{\beta}}$ is the vector of polynomial coefficients and $\mathrm{T}^{(\tilde{n})}(t) = [1\, t\, \ldots\, t^{\tilde{n}-1}]$. The polynomial coefficients are found as minimizers of the weighted least squares expression

$$\|W_{j+1,e}(t)(\boldsymbol{Y} - \mathrm{T}_{j+1,e}\boldsymbol{\beta})\|$$

for $\boldsymbol{\beta}$ satisfying the constraints (for $i \in e$)

$$\frac{\partial(\mathrm{T}^{(\tilde{n})}(t)\boldsymbol{\beta})}{\partial Y_i} \geq 0.$$

The weights, and hence the polynomial coefficients, depend on the argument t, through a moving kernel $K(t)$. $W_{j+1,e}(t)$ is a diagonal matrix of weights with elements $(W_{j+1,e})_{kk}(t) = K\left(\frac{t - t_{j,e,k}}{h_{j+1}}\right)$.

The constraints impose positivity of the elements in the prediction matrix. If $\tilde{n} \geq 1$, the rows of the matrix sum up to one, causing the elements to be convex. A bit confusingly from the terminology point of view, the constraint minimization problem itself is a non-convex combinatorial problem. As the problem is solved locally for all elements $t_{j+1,2k+1}$, this has limited computational consequences.

If the observations are not equidistant, then for the sake of smooth reconstructions after thresholding or other processing, the polynomial degree $\tilde{n} - 1$ should be at least one. That means that Nadaraya-Watson kernel estimation (for which $\tilde{n} = 1$) cannot possibly lead to smooth reconstructions. Indeed, if $\tilde{n} < 2$, the function $y = x$ cannot be represented in the scheme without nonzero details. As a consequence, the grid structure of the observations is reflected in a reconstruction where details have been thresholded.

4. ADAPTIVE MULTISCALE LOCAL POLYNOMIAL TRANSFORMS

4.1. Truncated kernel functions

For the **detection** of possible jumps during the transformation, we propose to compute for each coefficient three versions $\boldsymbol{w}^C$, $\boldsymbol{w}^L$ and $\boldsymbol{w}^R$ of the offsets as defined in (2). The offset $\boldsymbol{w}^C$ comes from the full kernel $K^C(t) = K(t)$ as introduced in Section 3.2, whereas the two others come from truncated kernels $K^L(t) = K(t) \cdot \chi_{t\leq 0}(t)$ and $K^R(t) = K(t) \cdot \chi_{t\geq 0}(t)$ for $\boldsymbol{w}^L$ and $\boldsymbol{w}^R$, respectively; see Figure 2. (In these definitions $\chi_A(t)$ stands for the characteristic or identicator function of a set A.) The idea is to choose the smallest of the three coefficients and store the choice for use in the reconstruction.

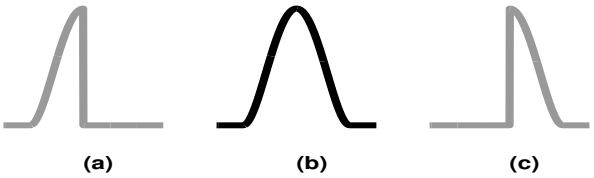

Fig. 2. Left truncated (a), full (b) and right truncated (c) kernels in adaptive kernel smoothing.

Once a truncated estimation is chosen, then truncation at that location should also be applied to adjacent offsets that have resulted from kernels whose supports contain the truncation point. This local update of coefficients is necessary for reasons of smooth reconstructions. It also avoids multiple

detections of the same break point. This results in an adaptive partitioning algorithm at each scale j, summarized below. Most of the administrative effort in item 4 is spent on finding the correct subinterval for a newly found partitioning point.

1. Compute $\boldsymbol{w}_j^C$, $\boldsymbol{w}_j^L$, and $\boldsymbol{w}_j^R$.
2. Initialize the partition of the grid locations $\mathcal{T}_j$ to the trivial partition
   $\mathcal{T}_j = \{T_{j,0}^0\}$, with $T_{j,0} = \{t_{j,1}, \ldots, t_{j,n_j}\}$,
   and n_j is the number of offsets (length of vectors $\boldsymbol{w}_j^s$ with $s \in \{L, C, R\}$).
3. Set $r = 0$
4. **repeat**
   - **find** k that maximizes
     $d_{j,k} = \max(|w_{j,k}^C| - |w_{j,k}^L|, |w_{j,k}^C| - |w_{j,k}^R|)$
   - **If** $|w_{j_k}^C| > \lambda$ (a threshold, see below) **and** $\max(|w_{j,k}^L|, |w_{j,k}^R|) < \lambda$
     then
     - **Find** p such that $t_{j,k} \in T_{j,p}^r$
     - **If** $|w_{j,k}^L| < |w_{j,k}^R|$,
       then $T_{j,p}^{r+1} = \{\ell \in T_{j,p}^r | \ell \leq k\}$
       else $T_{j,p}^{r+1} = \{\ell \in T_{j,p}^r | \ell \leq k-1\}$
     - Set $T_{j,p+1}^{r+1} = T_{j,p}^r \backslash T_{j,p}^{r+1}$,
       i.e. the complement of $T_{j,p}^{r+1}$ in $T_{j,p}^r$
     - **For** $q \in \{p, p+1\}$, compute $\boldsymbol{w}_{j,q}^C$, $\boldsymbol{w}_{j,q}^L$, and $\boldsymbol{w}_{j,q}^R$ and fill in into $\boldsymbol{w}_j^C$, $\boldsymbol{w}_j^L$, and $\boldsymbol{w}_j^R$
     - **For** $q < p$, set $T_{j,q}^{r+1} = T_{j,q}^r$
     - **For** $q > p$, set $T_{j,q+1}^{r+1} = T_{j,q}^r$
     - Set $\mathcal{T}_j = \{T_{j,p}^{r+1}, p = 0, \ldots, r\}$
     - Set r to $r+1$.

     else stop search = `TRUE`

   until stop search

4.2. Statistical testing

The routine described above compares the magnitude $|w_{j_k}^C|$ with the magnitudes $|w_{j_k}^L|$ and $|w_{j_k}^R|$. This can be seen as a statistical test of significance. Let $\mu_{j,k}^s = E(w_{j_k}^s)$ for $s \in \{L, C, R\}$, then we test if $H_0 : \mu_{j,k}^s = 0$ against $H_1 : \mu_{j,k}^C \neq 0$ and $\min(|\mu_{j,k}^L|, |\mu_{j,k}^R|) = 0$. The test can be repeated for every k separately, based on the test statistic $T_k = \max(|w_{j_k}^C|/|w_{j_k}^L|, |w_{j_k}^C|/|w_{j_k}^R|)$, or any equivalent value. This test statistic is independent from the variance of the coefficients. Its null distribution, however, is that of the maximum of two ratios, which is typically a heavy tailed variable. For instance, for Gaussian observations, the distribution under H_0 is a Cauchy variable. The heavy tails lead to tests with little power, many false positives or both.

In order to increase the power of the statistical tests, we estimate the variance of each coefficient based on all coefficients at scale j. This estimation proceeds in two steps. First, assuming that the observations are independent, identically distributed, we have for the covariance matrix $\Sigma_{\boldsymbol{y}} = \sigma^2 I$. From there we can, up to the unknown constant σ^2, find the structure of the covariance matrix of the coefficients using the recursion $\boldsymbol{s}_j = \widetilde{V}_j \boldsymbol{s}_{j+1}$, and the expression $\boldsymbol{w}_j = \widetilde{W}_j \boldsymbol{s}_{j+1}$. The corresponding covariance matrices are $\Sigma_{\boldsymbol{s}_j} = \widetilde{V}_j \Sigma_{\boldsymbol{s}_{j+1}} \widetilde{V}_j^T$ and $\Sigma_{\boldsymbol{w}_j} = \widetilde{W}_j \Sigma_{\boldsymbol{s}_{j+1}} \widetilde{W}_j^T$, allowing to standardize the detail coefficients as $w'_{j,k} = w_{j,k} \left[\left(\Sigma_{\boldsymbol{w}_j}\right)_{kk}\right]^{-1/2}$, for which we know that $\mathrm{var}(w'_{j,k}) = \sigma^2$. In a second step, the parameter σ can be estimated, using the sparsity of the decompositions. For Gaussian data, this could be, for instance, the median absolute deviation (MAD) based estimator

$$\widehat{\sigma}_j = \mathrm{median}(|\boldsymbol{w}'_j|)/\Phi^{-1}(3/4),$$

where $\Phi(x)$ is the cumulative Gaussian distribution, and $\Phi^{-1}(3/4) \approx 0.6745$. Based on this estimator, we can impose a threshold λ on $|\boldsymbol{w}'_j|/\widehat{\sigma}_j$ and select coefficients that are above the threshold, while at least one of its truncated equivalents are below the threshold.

4.3. Truncating and numerical condition

Truncation is incompatible with some of the techniques for numerically well conditioned transforms described above. Indeed, with a truncated kernel, all even neighbors used for prediction are taken from one side (left or right) of the point of evaluation. Unless the prediction is a least squares constant, such an extrapolation cannot possibly be a convex combination of the observations. On the other hand, as mentioned before, least squares constant, i.e., Nadaraya-Watson kernel prediction cannot possibly be smooth. Non-convex prediction remains fairly harmless as long as the update step satisfies a condition like the one in (5). This condition can, however, not be satisfied if the update step operates on one side of the detected singularity. If operates on both sides, it creates detail basis functions that stretch on both sides, while the corresponding coefficients are based on observations on one side only. This contradiction suggests that adaptive multiscale transforms might be better off without update steps. It should be brought in at this point that updates are necessary for good behavior near singularities, and this is exactly the place on which adaptivity focusses, be it in a different way.

5. RESULTS AND DISCUSSION

Figure 3 compares the outcome of a noise reduction using an updated multiscale local linear estimation with the outcome from an adaptive multiscale local linear estimation. The signal is observed with stationary additive noise, i.e., $\boldsymbol{Y} = \boldsymbol{f} + \boldsymbol{\varepsilon}$. The signal-to-noise ratios $\mathrm{SNR} = 10 \log_{10}(\|f - \overline{f}\|_2^2 / \mathrm{var}(\boldsymbol{\varepsilon}))$ for this example are 12.87 dB at the input, 24.07 dB for the non-adaptive output, and 24.61 dB for the adaptive alternative.

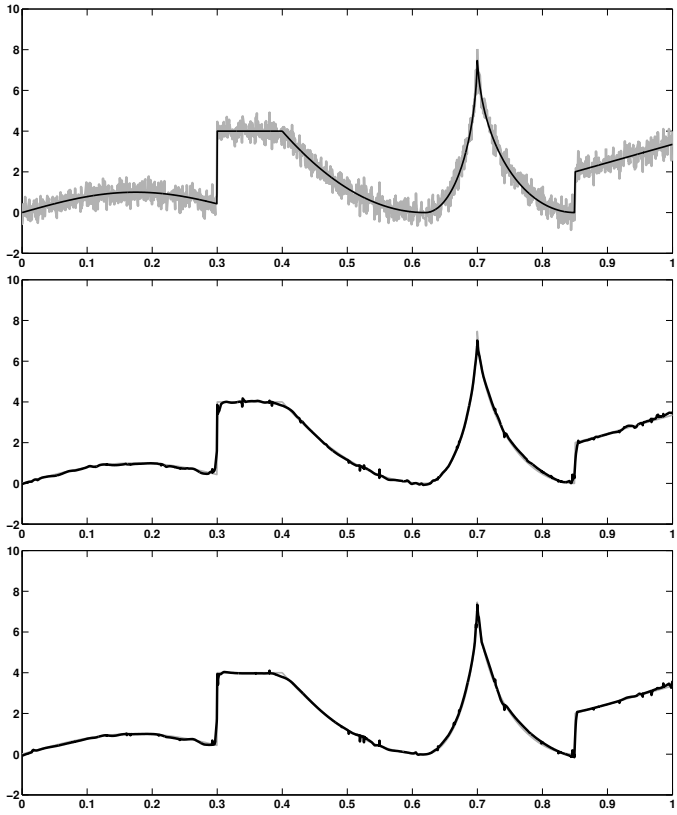

Fig. 3. (Top) Test signal with 2049 observations, subject to independent, homoscedastic, additive normal errors. (Middle) Reconstruction using updated multiscale local linear smoothing. (Bottom) Reconstruction with adaptive multiscale local linear smoothing.

The adaptive transform is computationally slightly more expensive, keeping, however, the linear order of complexity. The reconstruction from the adaptive transform is smoother in the intervals between the jumps, because the adaptive transform needs only one label per scale, indicating the location of the jump, while in a non-adaptive transform several large coefficients are grouped around this location, causing the the optimal threshold to be much lower. Another mechanism contributing to the potential of the adaptive transform is the variety of alternatives that can be considered when testing for the presence of a jump. In wavelet thresholding, the only alternative is to include or not a basis function. In adaptive transforms, the basis function can be included, half-included or otherwise taken into consideration.

6. REFERENCES

[1] M. Jansen, "Multiscale local polynomial smoothing in a lifted pyramid for non-equispaced data," *IEEE Transactions on Signal Processing*, vol. 61, no. 3, pp. 545–555, 2013.

[2] J. Fan and I. Gijbels, *Local Polynomial Modelling and its Applications*, Chapman and Hall, London, 1996.

[3] T. Cai and L.D. Brown, "Wavelet shrinkage for nonequispaced samples," *Annals of Statistics*, vol. 26, no. 5, pp. 1783–1799, 1998.

[4] A. Kovac and B. W. Silverman, "Extending the scope of wavelet regression methods by coefficient-dependent thresholding," *J. Amer. Statist. Assoc.*, vol. 95, pp. 172–183, 2000.

[5] A. Antoniadis and J. Fan, "Regularized wavelet approximations," *J. Amer. Statist. Assoc.*, vol. 96, no. 455, pp. 939–955, Sept. 2001.

[6] W. Van Aerschot, M. Jansen, and A. Bultheel, "Adaptive splitting for stabilizing 1-d wavelet decompositions," *Signal Processing*, vol. 86, no. 9, pp. 2447–2463, 2006.

[7] E. Vanraes, M. Jansen, and A. Bultheel, "Stabilizing wavelet transforms for non-equispaced data smoothing," *Signal Processing*, vol. 82, no. 12, pp. 1979–1990, Dec. 2002.

[8] S. K. Narang and A. Ortega, "Perfect reconstruction two-channel wavelet filter-banks for graph structured data," *IEEE Transactions on Signal Processing*, vol. 60, no. 6, pp. 2786–2799, 2012.

[9] P. J. Burt and E. H. Adelson, "Laplacian pyramid as a compact image code," *IEEE Trans. Commun.*, vol. 31, no. 4, pp. 532–540, 1983.

[10] M. Flierl and P. Vandergheynst, "An improved pyramid for spatially scalable video coding," in *Proc. IEEE Int. Conf. on Image Proc. — ICIP '05*, 2006, pp. 878–881.

[11] L. Liu, L. Gan, and T.D. Tran, "Lifting-based laplacian pyramid reconstruction schemes," in *Proc. IEEE Int. Conf. on Image Proc. — ICIP '08*, 2008, pp. 2812–2815.

[12] M. N. Do and M. Vetterli, "Framing pyramids," *IEEE Transactions on Signal Processing*, vol. 51, no. 9, pp. 2329–2342, 2003.

[13] M. Jansen, "Multiscale local polynomial models for estimation and testing," in *Topics in NonParametric Statistics*, M. Akritas, S. N. Lahiri, and D. Politis, Eds., p. To Appear. Springer, 2013.

[14] M. Jansen and P. Oonincx, *Second generation wavelets and applications*, Springer, 2005.