

In case the point to be interpolated is at the middle of two sampled points, the additional calculations reduce to the values given by Table II(c).

As can be seen from Table II, there is a tradeoff between the number of operations and accuracy of interpolation. If a definite intermediate value is to be calculated, the values of the kernel for that specific point may be kept in memory as a look-up table that reduces the number of calculations to a number that is comparable with the cardinal splines.

The kernel may be used for the sake of its precision if the data is oversampled since the number of necessary calculations is appreciably less than that of the fourth-order cardinal spline.

It may especially be used in case the data is to be multiplied with powers of two by using the interpolated values as if they were the sampled values. (It is suggested that the first multiplication should be obtained by using the fourth-order convolution kernel).

## V. CONCLUSIONS

A new fifth-order convolution interpolation kernel and its time-frequency domain properties have been introduced. The following results have been established:

- 1) This kernel gives better approximations when compared with the fourth-order convolution kernel and cubic cardinal spline interpolator if the data is oversampled.
- 2) The required number of calculations is slightly more than the third- and fourth-order convolution calculations and comparable with cubic spline interpolation calculations if look-up tables are used or points having relatively the same positions are calculated (the kernel needs not be recalculated). If midpoints are to be interpolated, a fifth-order local kernel needs slightly fewer calculations as compared with the third-order cardinal spline.
- 3) The results of interpolation calculations may be used to estimate the local sampling frequency.

## ACKNOWLEDGMENT

The author is very much indebted to the anonymous reviewers for their numerous suggestions, which have contributed a great deal to the contents of this correspondence. He is also very much indebted to Dr. K. Sayood and the associate editor for their editorial help.

## REFERENCES

- [1] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp. 1153-1160, Dec. 1981.
- [2] H. S. Hou and H. C. Andrews, "Cubic splines for image interpolation and digital filtering," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-26, pp. 508-517, Dec. 1978.
- [3] R. W. Schafer and L. R. Rabiner, "A digital signal processing approach to interpolation," *Proc. IEEE*, vol. 61, pp. 692-702, June 1973.
- [4] R. E. Crochiere and L. R. Rabiner, "Interpolation and decimation of digital signals—A tutorial review," *Proc. IEEE*, vol. 69, pp. 300-330, Mar. 1981.
- [5] M. Unser, A. Aldroubi, and M. Eden, "Fast B-spline transforms for continuous image representation and interpolation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, Mar. 1991.
- [6] ———, "Cardinal spline filters: Stability and convergence to the ideal sinc-interpolator," *Signal Processing*, vol. 28, pp. 127-138, Aug. 1992.
- [7] S. Roth, "Image processing: Past and present," *Lasers Optonics*, vol. 13, no. 7, July 1994.

## Deconvolution Filter Design for Fractal Signal Transmission Systems: A Multiscale Kalman Filter Bank Approach

Bor-Sen Chen and Wen-Sheng Hou

**Abstract**—A deconvolution filtering design is proposed for the  $1/f$  fractal signal transmission systems, with its design philosophy being based on multiscale Kalman deconvolution filter bank equipped in the analysis/synthesis wavelet filter bank. The role of wavelet transformation for  $1/f$  fractal signal process is exploited as a multiscale whitening filter for removing the properties of self-similarity and long-range dependence from the fractal signals.

## I. INTRODUCTION

The received signal in signal transmission systems can always be modeled as a convolution of transmitted signal and the impulse response of transmission channel under corrupted noise. The deconvolution task lies in restoring the input signal by removing the distortion of channel and corrupted noise. In most previous work, the signals in the deconvolution systems are assumed to be stationary processes. However, the  $1/f$  family of fractal signals are apparently nonstationary processes, and the conventional deconvolution filtering techniques are not easily applied toward removal of the distortion of the channel and the corruption of noise.

A new method for time-varying signal analysis, which is referred as to the wavelet transform, has come under investigation during the recent past [1], [2]. The signal system in practical engineering applications is designed with the capability of tackling a nonstationary signal or noise. Wornell [2], [3] has exploited the role of the wavelet transformation as a whitening filter that is effective in removing the properties of self-similarity and strong long-range dependence from  $1/f$  processes. However, the distortion of channel in transmission is neglected in their approach. The signal restoration in signal transmission systems has recently become an important topic in the application of wavelets and has been solved by Chen and Lin via the multiscale Wiener filtering method [8] from the frequency domain viewpoint. However, the undecimated coefficients of the multiscales are assumed to be white signal processes in their approach. In this correspondence, the signal reconstruction problem of fractal signal transmission systems is treated from multiscale Kalman filtering technique from the point of view of state space model; moreover, we propose an AR model to approach the colored processes of undecimated coefficients of multiscales.

At the beginning, the convolution of the signal transmission system is transformed into several multiscale convolutions in the time-scale domain via the analysis part of filter bank based on the orthonormal wavelet bases. The  $m$ th-scale convolution in the  $m$ th subband of the wavelet filter bank is observed here to be a convolution of the  $m$ th-scale input signal with the impulse response of transmission channel under the  $m$ th-scale corrupted noise. Then, the  $m$ th-scale Kalman deconvolution algorithm in the  $m$ th subband of the wavelet filter bank is employed to estimate the  $m$ th-scale transmitted signal.

Manuscript received January 25, 1996; revised December 9, 1996. This work was supported by National Science Council under Contract NSC 83-0404-E-007-042. The associate editor coordinating the review of this paper and approving it for publication was Dr. Zhi Ding.

The authors are with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu, Taiwan, R.O.C.

Publisher Item Identifier S 1053-587X(97)03334-5.

Consequently, while equipped in wavelet filter bank, the conventional Kalman optimal state estimation algorithm can be employed to treat this deconvolution problem for fractal signal transmission systems.

After the multiscales of transmitted signal have been estimated in each subband of the wavelet filter bank, a synthesis of input signal estimation is explored by the subsequent synthesis part of the wavelet filter bank. The deconvolution algorithm is consequently implemented in a parallelepiped structure. Finally, the performance of the deconvolution algorithm is discussed with several simulation examples given for illustrating the proposed method.

## II. PROBLEM FORMULATION AND A REVIEW OF MULTISCALE REPRESENTATION OF SIGNALS

### A. Problem Formulation

A signal transmission system is of the following form:

$$y(t) = \int_{-\infty}^{\infty} c(\tau)u(t-\tau)d\tau + w(t) \quad (1)$$

where

- $y(t)$  received signal,
- $c(t)$  impulse response of the transmission channel,
- $u(t)$  input signal,
- $w(t)$  additive noise.

In this signal transmission system, the channel impulse response  $c(t)$  is time invariant, the input signal  $u(t)$  is a  $1/f$  family of fractal signal, and the noise  $w(t)$  is a zero mean white gaussian noise. The restoration problem of signal transmission system (1) is to estimate  $u(t)$  ( $1/f$  signal) from  $y(t)$  with additive noise  $w(t)$ .

### B. A Review of Multiscale Represent of Signals

The approximated signal  $A_m u(t)$  of  $u(t)$  at the  $m$ th scale (resolution- $2^m$ ) [1] is

$$A_m u(t) = \sum_n [A_m^d u](n) \phi_m^n(t) \quad (2)$$

where  $\phi_m^n(t) = \phi(2^m t - n)$ , and  $\phi(t)$  is a basic scaling function.

The additional information or detail in going from a resolution- $2^m$  approximation  $A_m u(t)$  to a resolution- $2^{m+1}$  approximation  $A_{m+1} u(t)$  is

$$D_m u(t) = A_{m+1} u(t) - A_m u(t) = \sum_n [D_m^d u](n) \psi_m^n(t) \quad (3)$$

where  $\psi_m^n(t) = \psi(2^m t - n)$ , and  $\psi(t)$  is the basic wavelet. The approximate coefficients  $[A_m^d u](n)$  and the detailed coefficients  $[D_m^d u](n)$  can be obtained [1], [2] as

$$[A_m^d u](n) = \sum_k \tilde{h}(2n-k) [A_{m+1}^d u](k) \quad (4)$$

$$[D_m^d u](n) = \sum_k \tilde{g}(2n-k) [A_{m+1}^d u](k) \quad (5)$$

$$[A_{m+1}^d u](n) = \sum_k \{h(n-2k) [A_m^d u](k) + g(n-2k) [D_m^d u](k)\} \quad (6)$$

where  $h(n) = \int \phi_{-1}^0(t) \phi_0^n(t) dt$  and  $g(n) = \int \psi_{-1}^0(t) \psi_0^n(t) dt$ . Furthermore, let  $\tilde{h}(n)$  and  $\tilde{g}(n)$  be the mirror filter of  $h(n)$  and  $g(n)$ , i.e.,  $\tilde{h}(n) = h(-n)$  and  $\tilde{g}(n) = g(-n)$ .

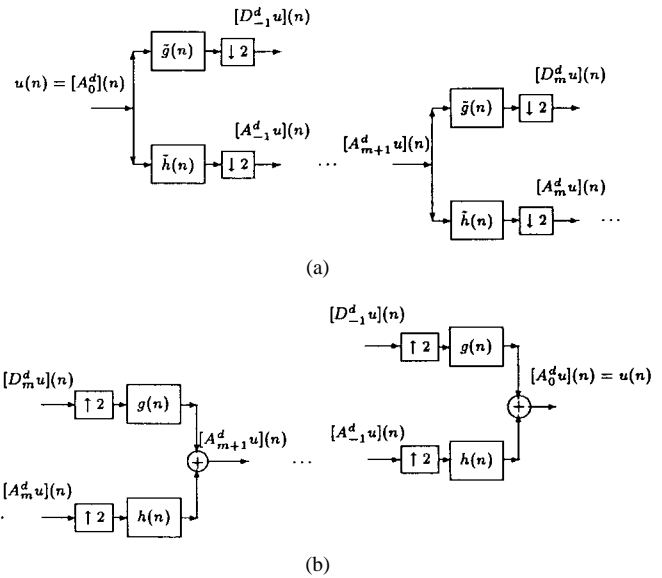


Fig. 1. (a) Decomposition of a discrete approximation  $[A_{m+1}^d u](n)$  into an approximation at a coarser resolution  $[A_m^d u](n)$  and the signal detail  $[D_m^d u](n)$ . (b) Reconstruction of a discrete approximation  $[A_{m+1}^d u](n)$  from an approximation at a coarse resolution  $[A_m^d u](n)$  and signal detail  $[D_m^d u](n)$ .

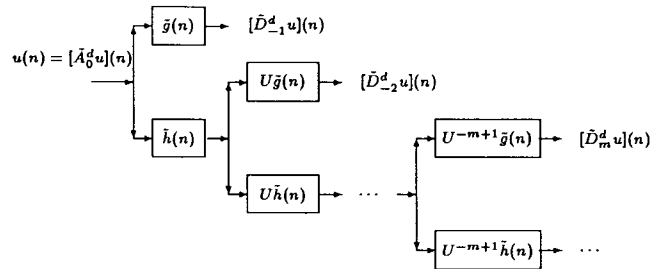


Fig. 2. (Undecimated) wavelet analysis filter bank.

*Remark 1:* If the discrete-time  $u(n)$  corresponds to samples  $[A_0^d u](n)$ , the analysis/synthesis equations (4)–(6) (or the algorithm in Fig. 1) extract the wavelet coefficients at scale  $m < 0$ .

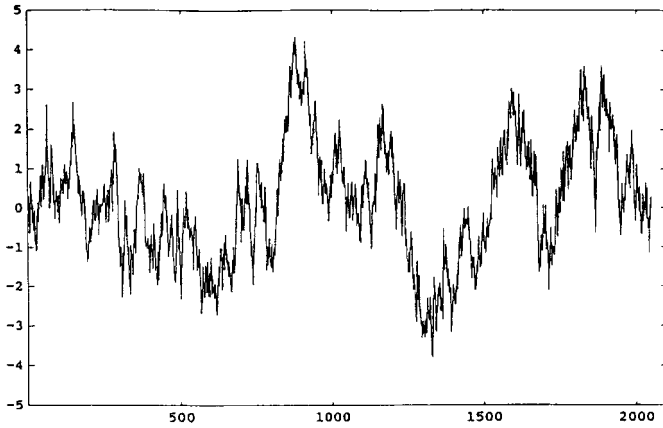
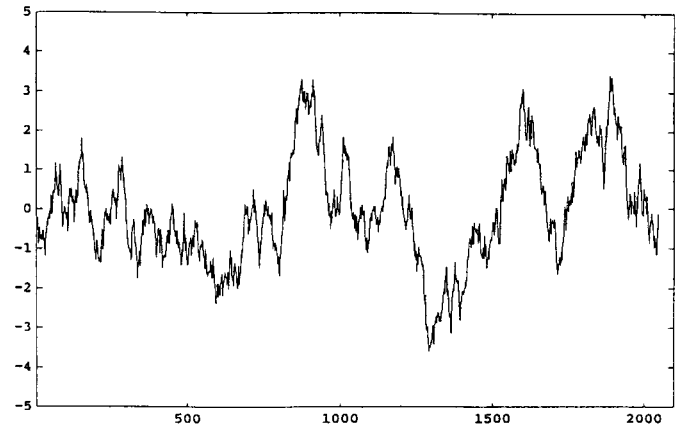
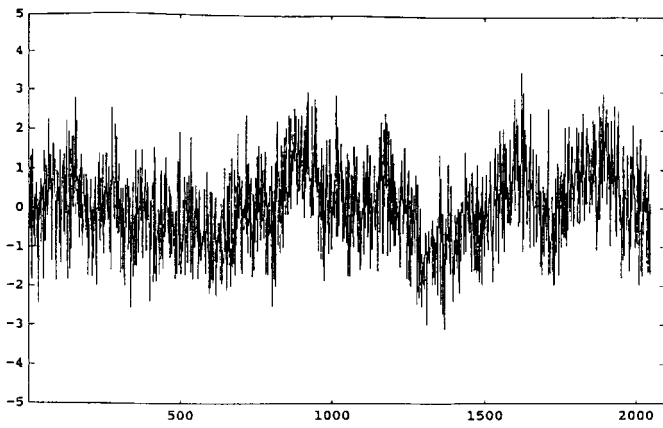
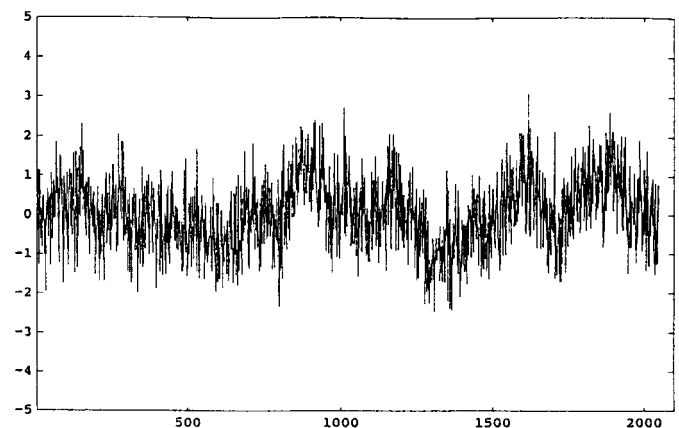
The wavelet transform discussed above, which can be denoted as the “decimated wavelet transform,” in Fig. 1 is not translation invariant. A new form of undecimated wavelet transform in Fig. 2 has been proposed [5], which is also useful for multiscale decomposition of deconvolution system. It is clear from [5] and [8] that downsampling  $[\tilde{D}_m^d u](n)$  by a factor  $2^{-m}$  precisely produces  $[D_m^d u](n)$ , that is

$$[\tilde{D}_m^d u](2^{-m}n) = [D_m^d u](n). \quad (7)$$

Note that the tilde  $[\tilde{\cdot}]$  is used to indicate the undecimated wavelet transform.

In our design, the received signal  $y(t)$  in Fig. 3 is transformed into several multiscale signals in each subband of wavelet filter bank based on the orthonormal wavelets. A multiscale deconvolution filter design in each subband of wavelet filter bank can therefore employ a conventional Kalman filter for estimating a multiscale transmitted signal in each subband of wavelet filter bank with accuracy. Once the estimation of multiscale transmitted signal in each subband of wavelet filter bank is finished, a synthesis of estimated transmitted signal  $\hat{u}(t)$  can then be explored by the subsequent wavelet synthesis filter bank.



Fig. 4. Transmitted 1/f fractal input signal  $u(t)$  with  $\gamma = 1.66$ .Fig. 6. Reconstructed signal  $\hat{u}(t)$  by the proposed algorithm in Example 1.Fig. 5. Received signal  $y(t)$  in Example 1.Fig. 7. Reconstructed signal  $\hat{u}(t)$  by conventional Kalman filter without multiscale decomposition in Example 1.

*Remark 2:* While the channel impulse response  $c(n)$  is of the IIR case, the system state variable  $\mathbf{X}_{-k}(n)$ ,  $\mathbf{V}_{-k}(n)$  and matrices  $\mathbf{A}_{-k}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  in (17) can also be modified equivalently.

The multiscale deconvolution filtering in the analysis (as in the above section) aims at obtaining the state estimate  $\hat{\mathbf{X}}_{-k}(n)$  from  $[\hat{D}_{-k}^d y](n)$  under the noise  $[\hat{D}_{-k}^d w](n)$  in (16). The conventional Kalman filtering algorithm [4] can be employed for estimating  $\hat{\mathbf{X}}_{-k}(n)$  to achieve the multiscale deconvolution of 1/f signal embedded in additive white noise, i.e.,

$$\begin{aligned} [\hat{D}_{-k}^d u](n) &= [1, 0, 0, \dots] \hat{\mathbf{X}}_{-k}(n|n) \\ &\text{for deconvolution filter} \end{aligned} \quad (18)$$

$$\begin{aligned} [\hat{D}_{-k}^d u](n - \tau) &= \underbrace{[0, 0, \dots, 0, 1, 0, \dots, 0]}_{\tau} \hat{\mathbf{X}}_{-k}(n|n) \\ &\text{for deconvolution smoother.} \end{aligned} \quad (19)$$

As an undecimated  $k$ th-scale optimal estimate of input signal  $[\hat{D}_{-k}^d u](n)$  is obtained, a decimated  $k$ th-scale optimal estimate of input signal  $[\hat{D}_{-k}^d u](n)$  can therefore be obtained by (7). Then, through the wavelet synthesis filter bank in Fig. 3, the deconvolution of fractal signal  $u(t)$  in the convolution system (1) is given by

$$\hat{u}(t) \approx \sum_{1 \leq k \leq M} \sum_n [\hat{D}_{-k}^d u](n) \psi_{-k}^n(t). \quad (20)$$

The summation of (20) are described in Fig. 3 on the basis of the wavelet filter bank in a parallelepiped structure.

## V. ILLUSTRATIVE SIMULATION RESULTS

Some numerical simulations are presented in this section by using artificial fractal signals [8] passing through a channel corrupted by a Gaussian noise. The transmitted fractal signal in the deconvolution system is shown in Fig. 4 with  $\gamma = 1.66$ . The scales  $M$  is set to 11.

The deconvolution filtering results obtained by the proposed algorithm are put into comparison with those of the other methods. For illustrating the performance of the proposed deconvolution algorithm for the case of filtering and smoothing, a performance is measured with SNR and SNR' defined in decibels as

$$\begin{aligned} \text{SNR} &= 10 \log_{10} \frac{\text{Var}(u(n))}{\text{Var}(u(n) - \hat{u}(n))} \\ \text{SNR}' &= 10 \log_{10} \frac{\text{Var}(y(n))}{\text{Var}(w(n))}. \end{aligned} \quad (21)$$

*Example 1:* The channel is modeled as an FIR filter with the following  $z$ -transform

$$\begin{aligned} C(z) &= -0.077 - 0.355z^{-1} + 0.059z^{-2} + z^{-3} \\ &\quad + 0.059z^{-4} - 0.273z^{-5}. \end{aligned}$$

The additive white Gaussian noise is added in this fractal signal with the SNR' being 0 dB. The distorted noisy fractal signal of channel output is shown in Fig. 5. The lag  $\tau$  is set to 5. The distorted noisy fractal signals are then processed using the proposed algorithm. A considerable amount of improvement is observed from the reconstructed signal in Fig. 6. It is obviously indicated from Fig. 7 to have a worse performance than the proposed algorithm if the

TABLE I  
COMPARISON OF SNR'S IN EXAMPLE 1

| SNR'     | Conventional Kalman Filter |                   | Multiscale Wiener Filter[8] |                   | Proposed Algorithm |                   |
|----------|----------------------------|-------------------|-----------------------------|-------------------|--------------------|-------------------|
|          | SNR<br>$\tau = 3$          | SNR<br>$\tau = 5$ | SNR<br>$\tau = 3$           | SNR<br>$\tau = 5$ | SNR<br>$\tau = 3$  | SNR<br>$\tau = 5$ |
| 0 dB     | 2.053 dB                   | 2.098 dB          | 8.189 dB                    | 9.396 dB          | 9.981 dB           | 10.114 dB         |
| 5 dB     | 3.580 dB                   | 5.508 dB          | 9.813 dB                    | 11.203 dB         | 12.084 dB          | 12.559 dB         |
| 10 dB    | 4.585 dB                   | 8.208 dB          | 9.996 dB                    | 12.013 dB         | 13.683 dB          | 14.875 dB         |
| 15 dB    | 5.045 dB                   | 9.964 dB          | 11.032 dB                   | 14.343 dB         | 14.557 dB          | 16.554 dB         |
| 20 dB    | 5.218 dB                   | 10.785 dB         | 12.098 dB                   | 15.623 dB         | 14.976 dB          | 17.549 dB         |
| $\infty$ | 5.513 dB                   | 10.966 dB         | 15.019 dB                   | 18.268 dB         | 15.338 dB          | 18.513 dB         |

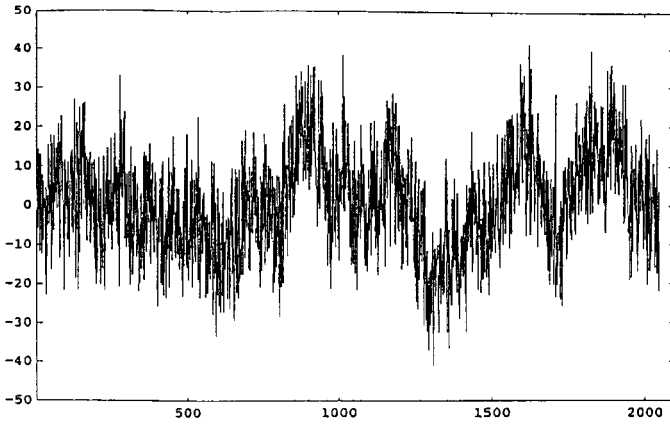


Fig. 8. Received signal  $y(t)$  in Example 2.

conventional Kalman filter is directly applied toward reconstructing the fractal signals from the channel output.

The comparisons among the conventional Kalman filter without multiscale decomposition, multiscale Wiener filter [8], and the proposed algorithm are shown in Table I.

Example 2: The channel is modeled as an IIR nonminimum-phase filter with the following  $z$ -transform:

$$C(z) = \frac{1 + 2.0z^{-1}}{1 - 0.5z^{-1}}. \quad (22)$$

The additive white Gaussian noise is added in this fractal with the SNR' being 0 dB. The distorted noisy fractal signal of channel output is shown in Fig. 8. The distorted noisy fractal signals are then processed using the proposed algorithm. The reconstructed signal is shown in Fig. 9, where the lag  $\tau$  is set to be 1. The reconstructed signals being directly transmitted by the conventional Kalman filter are confirmed by the results in Fig. 10. The fractal signal deconvolution problem is indicated by this figure to be incapable of being easily solved by the conventional Kalman filtering technique without using the multiscale decomposition technique.

The comparisons among the conventional Kalman filter without multiscale decomposition, multiscale Wiener filter [8], and the proposed algorithm are also shown in Table II. We have found that the proposed method has better performance than that of [8] because in [8], the undecimated coefficients of the multiscales are assumed to be white signal processes; however, in this paper, we have proposed an AR model to approach the colored processes of undecimated coefficients of multiscales.

A good reconstruction performance is achieved in the above simulation examples by the proposed multiscale deconvolution algorithms, even in very poor SNR cases. The reason for this is that the  $1/f$  processes at low frequencies (coarse scales) are dominant in energy, and the proposed multiscale deconvolution algorithm has exploited and used this important property of  $1/f$  nonstationary processes.

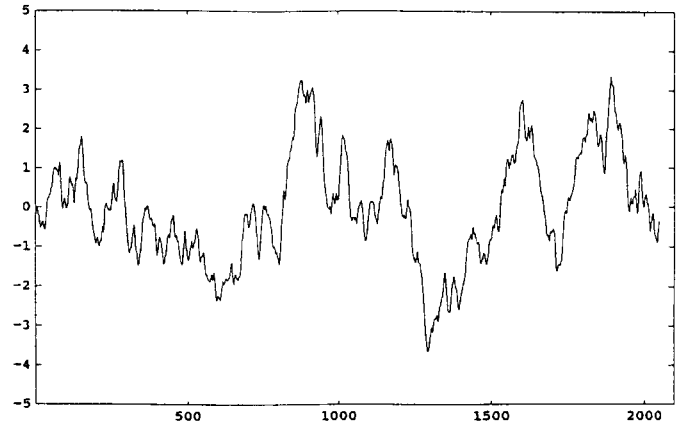


Fig. 9. Reconstructed signal  $\hat{u}(t)$  by the proposed algorithm in Example 2.

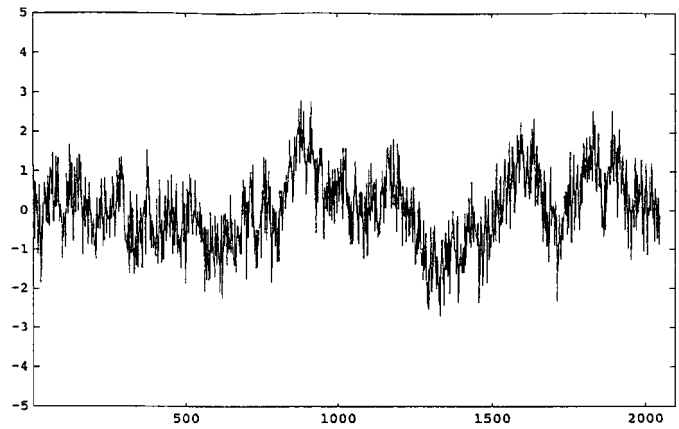


Fig. 10. Reconstructive signal  $\hat{u}(t)$  by conventional Kalman filter without multiscale decomposition in Example 2.

TABLE II  
COMPARISON OF SNR'S IN EXAMPLE 2

| SNR'     | Conventional Kalman Filter |                   | Multiscale Wiener Filter [8] |                   | Proposed Algorithm |                   |
|----------|----------------------------|-------------------|------------------------------|-------------------|--------------------|-------------------|
|          | SNR<br>$\tau = 3$          | SNR<br>$\tau = 5$ | SNR<br>$\tau = 3$            | SNR<br>$\tau = 5$ | SNR<br>$\tau = 3$  | SNR<br>$\tau = 5$ |
| 0 dB     | 4.173 dB                   | 4.345 dB          | 9.638 dB                     | 9.686 dB          | 10.203 dB          | 10.221 dB         |
| 5 dB     | 6.525 dB                   | 6.559 dB          | 11.775 dB                    | 11.831 dB         | 12.052 dB          | 12.076 dB         |
| 10 dB    | 7.360 dB                   | 7.377 dB          | 13.465 dB                    | 13.717 dB         | 13.926 dB          | 13.939 dB         |
| 15 dB    | 8.104 dB                   | 8.176 dB          | 15.388 dB                    | 15.808 dB         | 15.872 dB          | 15.873 dB         |
| 20 dB    | 9.652 dB                   | 9.845 dB          | 17.117 dB                    | 17.822 dB         | 17.924 dB          | 17.925 dB         |
| $\infty$ | 23.944 dB                  | 35.886 dB         | 34.226 dB                    | 37.093 dB         | 34.376 dB          | 37.205 dB         |

## VI. CONCLUSION

A wavelet analysis/synthesis filter bank is developed to equip with a multiscale Kalman deconvolution filter bank to solve the deconvolution problem of the fractal signal transmission systems. The wavelet filter bank plays the role of removing the self-similarity and long-range dependence of  $1/f$  fractal signal that cannot be easily treated by the conventional signal processing techniques. The Kalman deconvolution filter bank plays the role of removing the distortion of transmission channel and eliminating the effect of corrupted noise.

In order to improve the reconstruction performance, the multiscale Kalman smoother bank is also proposed. Since the proposed method is based on the Kalman filtering algorithm, it can be computed recursively and is more adequate for real-time signal reconstruction.

The proposed algorithm is confirmed from the illustrative simulation results to obtain a better performance than the conventional deconvolution method without the multiscale decomposition technique. Therefore, the proposed Kalman deconvolution filter bank can efficiently treat the deconvolution problem of nonstationary fractal signal transmission systems.

## REFERENCES

- [1] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 674–693, July 1989.
- [2] G. W. Wornell and A. V. Oppenheim, "Estimation of fractal signal from noisy measurement using wavelets," *IEEE Trans. Signal Processing*, vol. 40, pp. 611–623, Mar. 1992.
- [3] G. W. Wornell, "A Karhunen-Loève-like expansion for  $1/f$  processes via wavelets," *IEEE Trans. Inform. Theory*, vol. 36, pp. 859–861, July 1990.
- [4] J. M. Mendel, *Lessons in Digital Estimation Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1987.
- [5] M. J. Shensa, "The discrete wavelet transform: Wedding the À Troux and Mallat algorithm," *IEEE Trans. Signal Processing*, vol. 40, pp. 2464–2482, Oct. 1992.
- [6] A. H. Tewfik and M. Kim, "Correlation structure of the discrete wavelet coefficients of fractional Brownian motion," *IEEE Trans. Inform. Theory*, vol. 38, pp. 904–909, Mar. 1992.
- [7] P. Flandrin, "Wavelet analysis and synthesis of fractional Brownian motion," *IEEE Trans. Inform. Theory*, vol. 38, pp. 910–917, Mar. 1992.
- [8] B.-S. Chen and C. W. Lin, "Multiscale Wiener filter for the restoration of fractal signals: Wavelet filterbank approach," *IEEE Trans. Signal Processing*, vol. 40, pp. 2972–2982, Nov. 1994.
- [9] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*. Englewood Cliffs, NJ: Prentice-Hall, 1978.

### PWL Nonlinear Adaptive Filter via RLS and NLMS Algorithms

Nathalie Plaziac, Chon Tam Ledin, and Jean-Pierre Adoul

**Abstract**—The recursive least square (RLS) and the normalized least mean square (NLMS) algorithms are proposed for canonical piecewise linear (PWL) adaptive filters. The parameters are updated recursively in a manner similar to back-propagation. The simulation results indicate PWL adaptive filters can suitably model nonlinear systems.

#### INTRODUCTION

The recursive least square (RLS) algorithm is a fast convergent method for identifying the parameters of unknown systems. RLS converges faster than the normalized least mean square (NLMS) and least mean square (LMS) algorithms [4], [5]. On the other hand, the NLMS algorithm is more robust and less prone to numerical divergence than the RLS algorithm. Thus, the performances of both algorithms are of interest.

The RLS and NLMS algorithms have been widely used in linear adaptive filtering. They have also been used in nonlinear filtering to update the coefficients of some Volterra filters [6]. The nonlinear filters were reduced to a linear form in the coefficient space of the filters. Our problem is slightly different because the PWL filter coefficients cannot be decomposed in a linear manner. The back-propagation algorithm [7] and the linear decomposition problem suggests an interesting hybrid solution that uses the RLS and NLMS

Manuscript received February 3, 1994; revised December 9, 1996. The associate editor coordinating the review of this paper and approving it for publication was Dr. Virginia L. Stonick.

The authors are with the Department of Electrical Engineering, Université de Sherbrooke, Sherbrooke, P.Q., Canada J1K 2R1.

Publisher Item Identifier S 1053-587X(97)03351-5.

algorithms in a neural net form (see Fig. 1). In a related paper, Lin and Unbehauen [2] have used LMS in a nonneural net form.

The piecewise linear (PWL) functions are practical for two reasons: 1) They perform better and 2) require fewer coefficients than the Volterra filter to model strong nonlinearities [2]. Thus, the computational requirements are greatly reduced using an adaptive PWL function filter to approximate an unknown nonlinear filter. This note uses a canonical piecewise linear function of the form [1]

$$f(x(n)) = a + \vec{b}^T \vec{x}(n) + \sum_{j=1}^{\sigma} C_j |\vec{\alpha}_j^T \vec{x}(n) + \alpha_{jN}|. \quad (1)$$

Equation (1) is more general than the well-known back-propagation neural network. In fact, it can be suitable for weak or strong nonlinearity systems identification. Moreover, using finite sums, Sandberg [3] showed that the nonlinear part of (1) can approximate arbitrarily well any causal, time-invariant model that satisfies certain continuity and approximately-finite-memory conditions. Our approximation scheme is slightly more general than [2] since no symmetry condition is assumed, and the centers are allowed to be fitted to the origin.

The remaining sections are divided in three sections. Section I introduces the two new methods and provides their recursive algorithms. Section II presents the simulation results. Finally, Section III discusses the results along with some further research ideas.

#### I. ALGORITHM DEVELOPMENT

The proposed nonlinear filter is shown in Fig. 1 for system identification. The cost functions minimized are a) the exponentially weighted least square  $J(n) = \sum_{i=1}^n \lambda^{n-i} |e(i)|^2$  for RLS in which  $e(i)$  is the difference between the desired signal  $d(i)$  and the approximated signal  $f(x(i))$  and b) for NLMS, the squared euclidean norm  $J(n) = \|\vec{\omega}(n) - \vec{\omega}(n-1)\|^2$  subject to the constraint  $f(x(n)) = d(n)$  [4].  $\vec{\omega}(n)$  is the coefficient vector of the estimated linear model at time  $n$  [ $\vec{\omega}(n)$  would stand for  $\vec{\omega}_1(n)$  in (2) and  $\vec{\omega}_2(n)$  in (6)].

Rewriting (1) in terms of vectors and matrices,

$$y(n) = f(x(n)) = \vec{\omega}_1^T(n) \vec{X}(n). \quad (2)$$

The coefficient vector  $\vec{\omega}_1(n)$  is given by

$$\vec{\omega}_1(n) = [a(n) \quad \vec{b}^T(n) \quad C_1(n), \dots, C_{\sigma}(n)]^T. \quad (3)$$

The data vector  $\vec{X}(n)$  is given by

$$\vec{X}(n) = [1 \quad \vec{x}(n) \quad r_1(n), \dots, r_{\sigma}(n)]^T \quad (4)$$

where

$$r_i(n) = |\vec{\alpha}_i^T(n) \vec{x}(n) + \alpha_{iN}|. \quad (5)$$

Form (2) is linear in  $\vec{\omega}_1(n)$  and allows any linear algorithm for the parameters  $a(n)$ ,  $\vec{b}^T(n)$ , and  $C_i(n)$  ( $i = 1, \dots, \sigma$ ). Repeating this approach with  $\vec{\alpha}_i(n)$  and  $\alpha_{iN}$  ( $i = 1, \dots, \sigma$ ), (1) can be rewritten as

$$y(n) = B + \vec{C}^T \vec{\omega}_2(n) \quad (6)$$

where  $B$  is used as a bias for this step and is given by

$$B = a(n) + \vec{b}^T(n) \vec{x}(n). \quad (7)$$

We assume that the coefficient vector  $\vec{\omega}_1(n)$  has been adapted previously. Thus,  $\vec{C}$  is considered to be a data vector for this step

$$\vec{C} = [C_1 \operatorname{sgn}(\vec{\alpha}_1^T(n) \vec{x}(n) + \alpha_{1N}(n)) [\vec{x}^T(n), 1], \dots, C_{\sigma} \operatorname{sgn}(\vec{\alpha}_{\sigma}^T(n) \vec{x}(n) + \alpha_{\sigma N}(n)) [\vec{x}^T(n), 1]]^T. \quad (8)$$