PERFECT RECONSTRUCTING NONLINEAR FILTER BANKS

Dinei A. F. Florêncio and Ronald W. Schafer

Digital Signal Processing Laboratory
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332
floren@eedsp.gatech.edu rws@eedsp.gatech.edu

ABSTRACT

Perfect reconstruction (PR) filter banks have found numerous applications, and have received much attention in the literature. For *linear* filter banks, necessary and sufficient conditions for PR have been established for most practical situations. Recently, nonlinear filter banks have been proposed for image coding applications. These filters are generally simple, and produce better results than linear filters of same complexity. Nevertheless, the lack of general PR conditions limits these filters to cases where one of the filters is the identity. In this paper, we present a framework that allows, for the first time, the design of PR nonlinear filter banks including (non-trivial) filters on all channels. Although the framework does not include all nonlinear PR filter banks, it does include all previously published nonlinear filter banks, as well as all linear ones. This framework suggest new possibilities for the design of nonlinear PR filter banks.

1. INTRODUCTION

Perfect reconstruction filter banks (PRFBs) have found important applications in signal compression and analysis. In particular, critically decimated PRFBs play an important role in these applications, and have been extensively studied [1]. Most previous research concentrated on *linear* filter banks. In fact, for linear filter banks, necessary and sufficient conditions for perfect reconstruction (PR) have been established for most practical situations. Many techniques are also available for designing such systems. These results are based on analysis tools like z-transform, frequency domain analysis (Fourier transforms), and Shannon sampling theory. The lack of corresponding tools for nonlinear systems has hampered the development of perfect reconstruction nonlinear filter banks (PRNFBs), and early work on PRNFBs was restricted to non-critically decimated cases [2, 3, 4]. More recently, a better understanding of issues related to critical morphological sampling [5, 6] has allowed the introduction of critically decimated PRNFBs [7, 8]. As a general rule, these PRNFBs are less complex and produce better results than their linear counterparts. Nevertheless, in all previous critically decimated PRNFBs, the signal in one of the channels was produced by simply subsampling the original signal [7, 8, 9, 10].

In this paper we introduce a framework that allows, for the first time, the design of nonlinear filter banks including (non-trivial) filters on all channels. Although the framework may seem restricted at first sight, it does include all linear PRFBs as a particular case.

In Section 2 we explain why we focus on a certain class of nonlinear systems. These systems can be characterized as a composition of certain *elementary stages*, which are presented in Section 3. Section 4 shows some examples of PRNFBs based on the cascade of these stages. Section 5 discusses the generality of the proposed framework, and Section 6 presents the main conclusions.

2. RESTRICTING THE CLASS OF NONLINEAR SYSTEMS

The development of interesting results for linear systems is heavily dependent on the restriction that was put on them (namely, requiring them to be linear). Although we would like to relax the requirement of linearity, we still find it necessary to put some sort of restriction on the system in order to produce useful results. If we allow the nonlinear systems to be completely general, even some basic concepts can be affected. For example, observe that the meaning of *"critical decimation"* may change if we consider the following nonlinear system:

Example 1 *Given a signal (possibly sampled at the Nyquist rate) $x[n]$, produce a (four times lower sampling rate) signal $y[n]$ by interlacing the (decimal) digits of four samples of $x[n]$. For example, suppose $x[4] = 0.22222$, $x[5] = 0.34567$, $x[6] = 0.99999$, and $x[7] = 0.18181$. Then we have $y[1] = 0.23912498259126982791$. Perfect reconstruction of $x[n]$ from the (undistorted) samples of $y[n]$ is clearly possible by just inverting the process.*

In this example, we reduced the total sampling rate of a signal by a factor of four, and could still perfectly reconstruct the signal. Nevertheless, nothing was gained with the sampling rate reduction, since **y** would have to be represented at a four times higher precision. It is easy to note the kind of trouble this can bring to a completely general framework for critically decimated nonlinear systems.

This work supported in in part under the Joint Services Electronics Program, Contract DAAH-04-93-G-0027, and by CNPq (Brazil) under contract 200.246-90/9.

0-7803-3192-3/96 $5.00©1996 IEEE

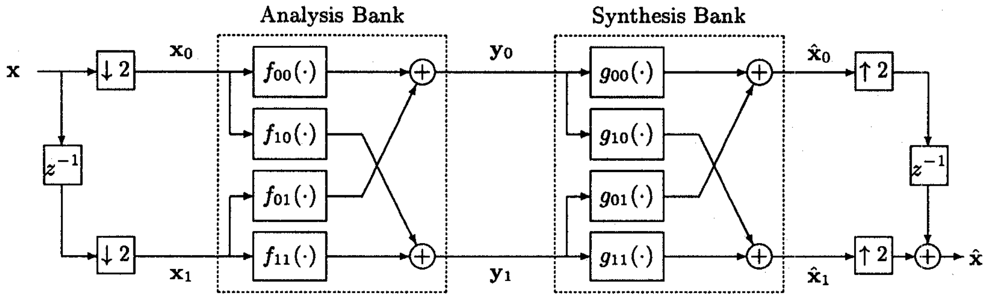

Figure 1: A polyphase two-channel nonlinear filter bank.

To avoid this kind of misleading transformation, we restrict our study to a class of filters that can be implemented as a cascade of certain *elementary stages*, which we describe in next section.

It is worthwhile to mention that this cascade of *elementary stages* is able to implement *any* critically decimated linear PRFB [11], as well as all the recently published critically decimated PRNFBs [7, 8, 9]. Moreover, it also allows for filtering both channels, in contrast to the previously published approaches for PRNFBs [7, 8, 9].

3. THE ELEMENTARY STAGES

Figure 1 depicts a polyphase implementation of a two-channel filter bank. It can be shown that most linear filter banks can be implemented in this form [1]. Although a PRNFB cannot in general be represented in polyphase form, we restrict our study to a subclass of PRNFBs which can be represented as a cascade of simplified polyphase stages (called elementary stages).

Restricting our notation to the two-channel filter bank depicted in Figure 1, we denote the two polyphase components of a signal $\mathbf{x}$ by $\mathbf{x}_0$ and $\mathbf{x}_1$, i.e., $x_0[n] = x[2n]$, and $x_1[n] = x[2n-1]$. The polyphase components of the transformed signal (also called *subbands* in linear PRFB) are $\mathbf{y}_0$ and $\mathbf{y}_1$, and can be expressed as:

$$\mathbf{y}_0 = f_{00}(\mathbf{x}_0) + f_{01}(\mathbf{x}_1), \text{ and} \quad (1)$$

$$\mathbf{y}_1 = f_{10}(\mathbf{x}_0) + f_{11}(\mathbf{x}_1), \quad (2)$$

where $f_{00}(\cdot)$, $f_{01}(\cdot)$, $f_{10}(\cdot)$, and $f_{11}(\cdot)$ are the (possibly nonlinear) polyphase *analysis* filters. The reconstructed signals $\hat{\mathbf{x}}_0$ and $\hat{\mathbf{x}}_1$ can be expressed as:

$$\hat{\mathbf{x}}_0 = g_{00}(\mathbf{y}_0) + g_{01}(\mathbf{y}_1), \text{ and} \quad (3)$$

$$\hat{\mathbf{x}}_1 = g_{10}(\mathbf{y}_0) + g_{11}(\mathbf{y}_1), \quad (4)$$

where $g_{00}(\cdot)$, $g_{01}(\cdot)$, $g_{10}(\cdot)$, and $g_{11}(\cdot)$ are the (possibly nonlinear) polyphase *synthesis* filters.

The PR condition for this system can be obtained by substituting equations (1) and (2) in (3) and (4), and requiring $\hat{\mathbf{x}} = \mathbf{x}$, yielding:

$$\mathbf{x}_0 = g_{00}(f_{00}(\mathbf{x}_0) + f_{01}(\mathbf{x}_1)) + g_{01}(f_{10}(\mathbf{x}_0) + f_{11}(\mathbf{x}_1)), \quad (5)$$

$$\mathbf{x}_1 = g_{10}(f_{00}(\mathbf{x}_0) + f_{01}(\mathbf{x}_1)) + g_{11}(f_{10}(\mathbf{x}_0) + f_{11}(\mathbf{x}_1)). \quad (6)$$

Using the above PR conditions, we now introduce the three classes of "elementary" stages that we have considered.

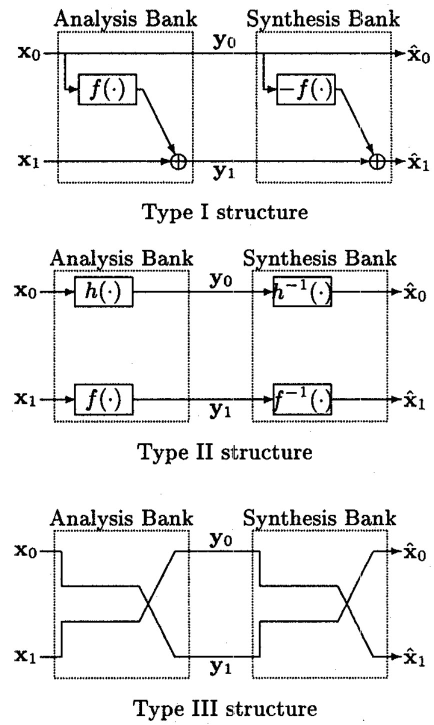

Figure 2: The three types of elementary stages.

Type I: (Hierarchical DPCM - like) A *type I structure* is one where $f_{00}(\cdot) = f_{11}(\cdot) = I$, (i.e., the identity transformation), and either $f_{10}(\cdot) = 0$ or $f_{01}(\cdot) = 0$. The other one can be any causal transformation (or any transformation that can be made causal by introducing a finite delay). Then PR can be obtained by making $g_{01}(\cdot) = -f_{01}(\cdot)$, $g_{10}(\cdot) = -f_{10}(\cdot)$, and $g_{00}(\cdot) = g_{11}(\cdot) = I$.

Type II: (DPCM - like) A *type II structure* is one where $f_{10}(\cdot) = f_{01}(\cdot) = 0$, and both $f_{00}(\cdot)$ and $f_{11}(\cdot)$ are invertible functions. In this case, PR can be obtained by making $g_{01}(\cdot) = g_{10}(\cdot) = 0$, $g_{00}(\cdot) = f_{00}^{-1}(\cdot)$, and $g_{11}(\cdot) = f_{11}^{-1}(\cdot)$.

Type III: (channel swapping) A *type III structure* is one where $f_{10}(\cdot) = f_{01}(\cdot) = I$, and $f_{00}(\cdot) = f_{11}(\cdot) = 0$. In this case, PR can be obtained by making $g_{01}(\cdot) = g_{10}(\cdot) = I$, and $g_{00}(\cdot) = g_{11}(\cdot) = 0$.

Figure 3: Using the cascade to reduce aliasing effects: (a) original; (b)(c) lower and upper bands, as in [10]; (d) re-filtered lower band; (e) linear with same window size; (f) linear with bigger filter. (note that (b) through (f) have been interpolated for plotting)

We note that if we restrict the functions $f_i(\cdot)$ to be linear, and use the usual matrix notation, a Type I stage corresponds to a triangular matrix with all ones in the diagonal, a Type II stage corresponds to a diagonal matrix, and a Type III stage corresponds to a permutation matrix.

4. CASCADE AND RECURSION OF ELEMENTARY STAGES

Since each elementary stage allows PR, it is clear that if a series of stages is cascaded, the composed system still allows PR. The following simple example shows how different types of elementary systems can be cascaded to produce PRNFBs that include operators on both channels. (as opposed to the single-channel filtering proposed in [7, 8, 9, 10])

Example 2 *Given a signal* $\mathbf{x}$ *such that* $x[n] > 0, \ \forall n \in N$, *consider the following cascade of elementary stages:*

stage 1: $f_{00}(x) = f_{11}(x) = x$, $f_{01}(x) = 0$, $f_{10}(x) = x^2$.

stage 2: $f_{00}(x) = f_{11}(x) = x$, $f_{01}(x) = -log(x)$, *and,* $f_{10}(x) = 0$.

stage 3: $f_{00}(x) = x$, $f_{11}(x) = \sqrt{x}$, *and,* $f_{01}(x) = f_{10}(x) = 0$.

Now, since each of these stages is invertible, so is the composed (cascade) system. The composed filter will be:

$$y_0[n] = x_0[n] - \log((x_0[n])^2 + x_1[n]), \quad (7)$$

$$y_1[n] = \sqrt{(x_1[n])^2 + x_0[n]} \quad (8)$$

The inverse system can be easily computed by applying the inverse filter of each stage in the reverse order, and will be:

$$x_0[n] = y_0[n] + 2\log(y_1[n]), \quad (9)$$

$$x_1[n] = (y_1[n])^2 - (y_0[n] + 2\log(y_1[n]))^2. \quad (10)$$

PRNFBs have been successfully applied to image and video coding [7, 8, 10, 12]. The PRNFBs used in all these works is based on a hierarchical cascade of a type I stage. In other words, at each level, one of the subsignals is obtained by simply subsampling the original signal. Although this has produced good coding results, this simple subsampling may introduce patterns not present in the original signal. This is particularly true when subsampling images with strong regular patterns or texture. A typical example is the image "Barbara". We selected a section of the decomposition of that image where this effect is most pronounced. Figure 3-a shows the original image at that level, while Figure 3-b shows the subsampled version, as used in [7, 8, 10]. Although the extraneous pattern that appears in the subsampled image does not affect the coding, it can limit the use of the lower resolution images for some applications (e.g., scaling). Until now, the simple subsampling was a condition to guarantee PR, and nothing could be done to reduce those patterns.

In next example, we show that we can use a cascade of elementary stages to reduce those patterns, while still preserving the PR property.

Example 3 *In this example, the signal is a 2-D signal in a rectangular grid, and the 2:1 subsampling maps the original signal into two quincunx subsignals (see [9]). The first stage is the same as described in [9], i.e., a type I structure, where* $f_{10}(\cdot)$ *is a 4-point median filter, and* $f_{01}(\cdot) = 0$. *The second stage is also a type I structure, but with* $f_{10} = 0$ *and* f_{01} *as a 4-point median followed by a 0.5 constant gain.*

The objective of the second stage is to reduce the aliasing effects due to the non-filtering of the $\mathbf{y}_0$ at the first stage. Figures 3-b and 3-c show the two subsignals after the first stage. Figure 3-d shows the result of the lower band after the second filtering stage. Note the reduction in the extraneous frequency when compared to Figure 3-b. Remember that the information in Figures 3-d and 3-c together is sufficient to perfectly reconstruct the signal.

It should also be pointed out that aliasing cancelation is a much easier task on linear PRFB, where we can make use of frequency domain analysis to design the filter. Figures 3-e and 3-f show the result of using linear filters. In 3-e the filter uses the same window size as the nonlinear filter, and it can be seen that similar results are obtained. Nevertheless, the compression performance of linear PRFB is usually worse for similar complexity.

It is also possible to apply a new PRNFB to one (or more) of the channels, e.g., to produce a multiresolution

Figure 4: A hierarchical nonlinear decomposition.

decomposition similar to that used in wavelet decomposition. Figure 4 shows an example of applying the system described in Example 3 in a hierarchical fashion.

Finally, we would like to mention that the problem of *designing* PRNFB is far from solved. Although the proposed framework opens important new possibilities, the lack of a frequency domain and other design tools still limits the complexity of nonlinear filters that can be designed. We are currently investigating the use of adaptive morphological filters [13] for this purpose.

5. HOW GENERAL IS THE FRAMEWORK?

The elementary stages we have considered are quite simple. It is also clear that many PRNFBs do not fit into the framework, i.e., cannot be decomposed into elementary stages (e.g., the system described in Example 1). This might give the impression that the framework is overly restricted. Nevertheless, we note that any linear PRFB can be decomposed into elementary stages.

To decompose a linear PRFB into elementary stages, we first find the polyphase representation of the system. A single-stage polyphase representation always exists for a linear system (the same is not true for nonlinear systems). The system can then be represented by the corresponding polyphase matrix $\mathbf{A}$, such that $\mathbf{y} = \mathbf{A}\mathbf{x}$, and the PR condition is equivalent to requiring that $\mathbf{A}$ be non-singular. But if $\mathbf{A}$ is non-singular, then it can be decomposed into a diagonal matrix (its Smith-McMillan form) by a series of elementary row/column operations. Each of these operations correspond to type I or type III elementary stages, and the diagonal matrix corresponds to a type II stage. Therefore, any linear PRFB can be decomposed into a cascade of elementary stages.

6. CONCLUSIONS

In this paper we present a sufficient condition for perfect reconstruction in nonlinear critically decimated filter banks. Although it is not a necessary condition, it is general enough to include all *linear* perfect reconstruction filter banks, as well as all previously published nonlinear filter banks.

The framework presented in this paper opens new possibilities for the analysis and design of nonlinear filter banks. This should find important application in filter design for traditional applications, like image coding and signal analysis. Besides, including nonlinear transformations in PRFBs may open new areas. For example, motion compensation (a nonlinear technique) can be included in the same framework as the linear transforms usually used for transforming the residual. An early application of this is reported in [12].

Finally, it should be mentioned that better techniques are still needed for nonlinear filter design. We hope the framework presented in this paper will encourage their development.

7. REFERENCES

[1] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Prentice-Hall, 1993.

[2] F.-K. Sun and P. Maragos, "Experiments on image compression using morphological pyramids," in *VCIP'89*, pp. 1303–1312, 1989.

[3] A. Toet, "A morphological pyramidal image decomposition," *Pattern Recog. Lett.*, vol. 9, pp. 255–261, May 1989.

[4] Z. Zhou and A. N. Venetsanopoulos, "Morphological methods in image coding," in *ICASSP'92*, pp. 481–484, Vol. III, 1992.

[5] D. A. F. Florêncio and R. W. Schafer, "Critical morphological sampling and applications to image coding," in *ISMM'94*, Sept. 1994.

[6] —, "Critical morphological sampling, part 2: Gray level signals." under preparation.

[7] —, "A non-expansive pyramidal morphological image coder," in *ICIP'94*, vol. 2, pp. 331–334, Nov. 1994.

[8] O. Egger and W. Li, "Very low bit rate image coding using morphological operators and adaptive decompositions," in *ICIP'94*, vol. 2, pp. 326–330, Nov. 1994.

[9] R. de Queiroz and D. A. F. Florêncio, "A pyramidal coder using a nonlinear filter bank," in *ICASSP'96*, 1996.

[10] R. de Queiroz, D. A. F. Florêncio, and R. W. Schafer, "Non-expansive pyramid for image coding using a nonlinear filter bank", under review, IEEE Trans. on Image Processing.

[11] F. A. M. L. Bruekers and A. W. M. van den Enden, "New networks for perfect inversion and perfect reconstruction," *IEEE Journal on Selected Areas in Comm.*, vol. 10, pp. 130–137, Jan. 1992.

[12] R. Armitano, D. A. F. Florêncio, and R. W. Schafer, "The motion transform: A new motion compensation technique," in *ICASSP'96*, 1996.

[13] L. F. C. Pessoa and P. Maragos, "Morphological/rank neural networks and their adaptive optimal design for image processing," in *ICASSP'96*, 1996.