Symmetric-Extension-Compatible Reversible Integer-To-Integer Wavelet Transforms

Michael D. Adams*, *Member, IEEE* and Rabab K. Ward, *Fellow, IEEE*

Abstract

Symmetric extension is explored as a means for constructing nonexpansive reversible integer-to-integer (ITI) wavelet transforms for finite-length signals. Two families of reversible ITI wavelet transforms are introduced, and their constituent transforms are shown to be compatible with symmetric extension. One of these families is then studied in detail, and several interesting results concerning its member transforms are presented. In addition, some new reversible ITI structures are derived that are useful in conjunction with techniques like symmetric extension. Lastly, the relationship between symmetric extension and per-lifting-step extension is explored, and some new results are obtained in this regard.

EDICS Category: 3-COMP (Signal Representation and Compression)

***Corresponding author:**

Michael D. Adams

Dept. of Electrical and Computer Engineering

University of Victoria

P.O. Box 3055 STN CSC

Victoria, BC, V8W 3P6, Canada

Tel: 1-250-721-6025, Fax: 1-250-721-6052

E-mail: `mdadams@ece.uvic.ca`

Other author:

Rabab K. Ward

Dept. of Electrical and Computer Engineering

University of British Columbia

2356 Main Mall

Vancouver, BC, V6T 1Z4, Canada

Tel: 1-604-822-6894, Fax: 1-604-822-9013

E-mail: `rababw@ece.ubc.ca`

Symmetric-Extension-Compatible Reversible Integer-To-Integer Wavelet Transforms

Michael D. Adams, *Member, IEEE*, and Rabab K. Ward, *Fellow, IEEE*

Abstract

Symmetric extension is explored as a means for constructing nonexpansive reversible integer-to-integer (ITI) wavelet transforms for finite-length signals. Two families of reversible ITI wavelet transforms are introduced, and their constituent transforms are shown to be compatible with symmetric extension. One of these families is then studied in detail, and several interesting results concerning its member transforms are presented. In addition, some new reversible ITI structures are derived that are useful in conjunction with techniques like symmetric extension. Lastly, the relationship between symmetric extension and per-lifting-step extension is explored, and some new results are obtained in this regard.

I. Introduction

During the last several years, reversible integer-to-integer (ITI) wavelet transforms [1–3] have become a popular tool for use in signal coding applications, especially image compression (e.g., [4–9]). Such transforms are invertible in finite-precision arithmetic (i.e., reversible), map integers to integers, and serve as nonlinear approximations to conventional (i.e., linear) wavelet transforms [10]. Frequently, reversible ITI wavelet transforms are defined in such a way as to operate on signals of infinite length. In practice, however, we almost invariably deal with finite-length signals. Therefore, we require some means for adapting reversible ITI wavelet transforms to handle such signals. In essence, we must address the well-known boundary problem that can arise when any finite-length signal is filtered (as filtering is part of the wavelet transform process). Furthermore, in signal coding applications, we almost always want to employ a transform that is nonexpansive (i.e., maps a signal of length N to a signal of length no greater than N). Therefore, we seek a solution to the boundary problem that yields nonexpansive transforms.

A similar boundary problem also arises in the case of conventional wavelet transforms, and numerous solutions have been proposed for this case over the years. Unfortunately, many of these methods are not easily applied or even useful in the reversible ITI case, as this case is complicated by both the nonlinear nature of the transforms involved and the requirement that reversibility be maintained.

In this manuscript, we explore symmetric extension [11, 12] as a means for constructing nonexpansive

This work was supported by the Natural Sciences and Engineering Research Council of Canada.

reversible ITI wavelet transforms. Although commonly used in the case of conventional wavelet transforms, symmetric extension is more difficult to apply in the reversible ITI case. In our work, we explain how this method can be utilized in the latter case. In addition, we consider the relationship between symmetric extension and another extension method which we refer to as per-lifting-step extension.

The remainder of this manuscript is structured as follows. We begin, in Section II, by presenting the basic notation and terminology used herein, and then proceed to discuss a few fundamentals of reversible ITI wavelet transforms in Section III. Next, our attention shifts to symmetric extension. In Section IV, we introduce this particular method and explain the conditions that must be satisfied in order for it to be applicable. In Section V, we present two families of reversible ITI wavelet transforms, and then proceed to demonstrate, in Section VI, that the transforms from these families are compatible with symmetric extension (i.e., can be used with symmetric extension in order to handle finite-length signals in a nonexpansive manner). In Section VII, we study one of the two transform families more closely, and present several interesting results concerning its constituent transforms. Afterwards, Section VIII proposes some new reversible ITI structures that are useful in conjunction with techniques like symmetric extension. In Section IX, we briefly introduce per-lifting-step (PLS) extension, and consider the relationship between PLS extension and symmetric extension. Finally, we conclude, in Section X, with a summary of our results and some closing remarks.

One particularly novel aspect of our work is our careful consideration of rounding operators and their properties. Our close attention to rounding operators has led to many new insights, allowing us to develop some entirely new ideas as well as extend some previously proposed ones. For example, through such insights, we were able to extend a previously existing transform family in order to yield one of the two families presented in this manuscript. This was possible, in part, due to a better understanding of why the base filter bank for the original family preserves certain signal symmetries. Also, we were able to derive several new reversible ITI structures (which are useful in conjunction with symmetric extension), and provide an example of a transform utilizing these structures.

II. Notation and Other Preliminaries

Before proceeding further, a short digression concerning the notation used in this manuscript is appropriate. The symbols $\mathbb{Z}$ and $\mathbb{R}$ denote the sets of integer and real numbers, respectively. Matrix and vector quantities are indicated using bold type. The symbols $\mathbf{I}_N$ and $\mathbf{J}_N$ denote the $N \times N$ identity and anti-identity matrices, respectively, where the subscript N may be omitted when clear from the context.

In the case of matrix multiplication, we define the product notation as follows:

$$\prod_{i=M}^{N} \mathbf{A}_i \triangleq \begin{cases} \mathbf{A}_N \mathbf{A}_{N-1} \cdots \mathbf{A}_{M+1} \mathbf{A}_M & \text{for } N \geq M \\ \mathbf{I} & \text{otherwise.} \end{cases}$$

For a Laurent polynomial $P(z)$, we denote the degree of $P(z)$ as $\deg P(z)$, and define this quantity as follows. In the case that $P(z)$ has the form $P(z) = \sum_{i=M}^{N} p_i z^{-i}$ where $N \geq M$, $p_M \neq 0$, and $p_N \neq 0$ (i.e., $P(z) \not\equiv 0$), $\deg P(z) \triangleq N - M$. In the case that $P(z)$ is the zero polynomial (i.e., $P(z) \equiv 0$), we define $\deg P(z) \triangleq -\infty$. Thus, for any two Laurent polynomials, $A(z)$ and $B(z)$, we have that $\deg(A(z)B(z)) = \deg A(z) + \deg B(z)$.

The notation $\mathcal{Z}$ and $\mathcal{Z}^{-1}$ are used to denote the forward and inverse Z transform operators, respectively. A signal $x[n]$ is said to have symmetry if it satisfies

$$x[n] = (-1)^S x[2C - n] \quad \text{for all } n \in \mathbb{Z},$$

for some choice of symmetry center C and symmetry type S, where C is an integer multiple of $\frac{1}{2}$ and $S \in \{0, 1\}$. If S is zero, $x[n]$ is said to be symmetric; if S is one, $x[n]$ is said to be antisymmetric.

For $\alpha \in \mathbb{R}$, the notation $\lfloor \alpha \rfloor$ denotes the largest integer not more than α (i.e., the floor function), and the notation $\lceil \alpha \rceil$ denotes the smallest integer not less than α (i.e., the ceiling function). The biased floor, biased ceiling, truncation, biased truncation, and rounding-away-from-zero (RAFZ) functions are defined, respectively, as

$$\operatorname{bfloor} \alpha \triangleq \left\lfloor \alpha + \tfrac{1}{2} \right\rfloor, \quad \operatorname{bceil} \alpha \triangleq \left\lceil \alpha - \tfrac{1}{2} \right\rceil,$$

$$\operatorname{trunc} \alpha \triangleq \begin{cases} \lfloor \alpha \rfloor & \text{for } \alpha \geq 0 \\ \lceil \alpha \rceil & \text{for } \alpha < 0, \end{cases} \quad \operatorname{btrunc} \alpha \triangleq \begin{cases} \operatorname{bfloor} \alpha & \text{for } \alpha \geq 0 \\ \operatorname{bceil} \alpha & \text{for } \alpha < 0, \end{cases} \quad \text{and} \tag{1}$$

$$\operatorname{rafz} \alpha \triangleq \begin{cases} \lceil \alpha \rceil & \text{for } \alpha \geq 0 \\ \lfloor \alpha \rfloor & \text{for } \alpha < 0, \end{cases} \tag{2}$$

where $\alpha \in \mathbb{R}$. One can show that, for all $\alpha \in \mathbb{R}$, the following relationships hold:

$$\lceil \alpha \rceil = -\lfloor -\alpha \rfloor \quad \text{and} \tag{3}$$

$$\operatorname{bceil} \alpha = \begin{cases} \operatorname{bfloor}(\alpha) - 1 & \text{if } \alpha \text{ is an odd integer multiple of } \frac{1}{2} \\ \operatorname{bfloor} \alpha & \text{otherwise.} \end{cases} \tag{4}$$

(For a proof of the latter, see [3].) The mod and signum functions are defined, respectively, as

$$\operatorname{mod}(x,y) \triangleq x - y\lfloor x/y \rfloor \quad \text{and} \quad \operatorname{sgn}\alpha \triangleq \begin{cases} 1 & \text{for } \alpha > 0 \\ 0 & \text{for } \alpha = 0 \\ -1 & \text{for } \alpha < 0, \end{cases}$$

where $x, y \in \mathbb{Z}$ and $\alpha \in \mathbb{R}$. In passing, we note that the mod function simply computes the remainder when x is divided by y, with division being defined in such a way that the remainder is always nonnegative. From (3) and the definition of the mod function, we trivially have the identities

$$\lfloor x/y \rfloor = \tfrac{x - \operatorname{mod}(x,y)}{y} \text{ and } \lceil x/y \rceil = \tfrac{x + \operatorname{mod}(-x,y)}{y} \quad \text{for } x, y \in \mathbb{Z} \text{ and } y \neq 0. \tag{5}$$

A rounding operator Q is said to be integer-bias invariant if

$$Q(\alpha + x) = Q(\alpha) + x \quad \text{for all } \alpha \in \mathbb{R} \text{ and all } x \in \mathbb{Z}.$$

Similarly, a rounding operator Q is said to be odd if

$$Q(\alpha) = -Q(-\alpha) \quad \text{for all } \alpha \in \mathbb{R}.$$

One can show that a rounding operator cannot be both odd and integer-bias invariant [13]. In passing, we note that the floor, biased floor, ceiling, and biased ceiling functions are integer-bias invariant (but not odd), while the truncation, biased truncation, and RAFZ functions are odd (but not integer-bias invariant). All rounding operators considered in this manuscript are tacitly assumed to be memoryless, time invariant, and leave integer values unchanged. Any (reasonable) rounding operator will preserve signal symmetry (but not necessarily signal antisymmetry), while any odd rounding operator will preserve both signal symmetry and antisymmetry.

III. Reversible ITI Wavelet Transforms

The reversible ITI wavelet transforms considered in this manuscript are based on the lifting framework [1, 2, 14, 15] and are associated with a uniformly-maximally-decimated (UMD) filter bank of the form shown in Fig. 1. In the figure, the analysis and synthesis polyphase filtering networks each consist of 2λ lifting-step filters $\{A_k\}_{k=0}^{2\lambda-1}$ and 2λ rounding operators $\{Q_k\}_{k=0}^{2\lambda-1}$. In the absence of the rounding operators (i.e., the $\{Q_k\}$), the system in Fig. 1 becomes a linear system, which corresponds to a conventional wavelet transform. Furthermore, this linear system can be realized using a UMD filter bank in its canonical form as shown in Fig. 2, where the $\{H_k\}_{k=0}^{1}$ and $\{F_k\}_{k=0}^{1}$ are analysis and synthesis filters, respectively. To convert a system of the form shown in Fig. 2 to the form shown in Fig. 1, one needs to solve the matrix factorization problem treated in [16, 17].

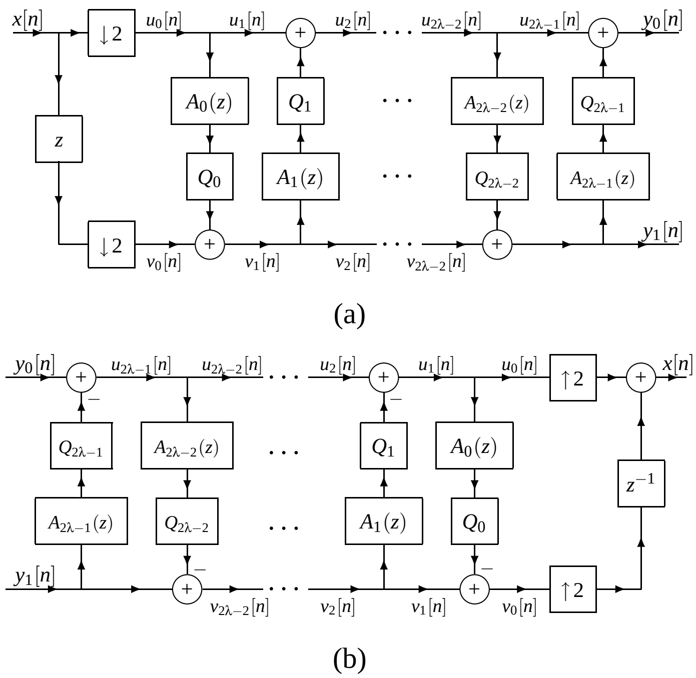

Fig. 1. Lifting realization of a reversible ITI wavelet transform. (a) Analysis filter bank and (b) synthesis filter bank.

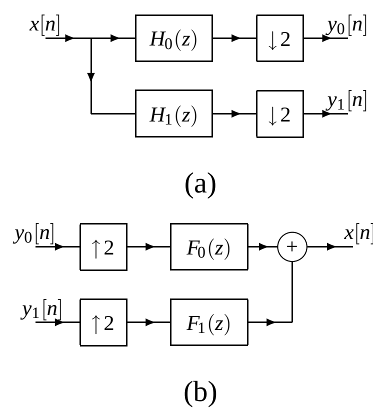

Fig. 2. The canonical realization of a conventional wavelet transform. (a) Analysis filter bank and (b) synthesis filter bank.

Perhaps it is worth nothing that, for the purposes of this manuscript, the term "wavelet transform" is used to mean a transform associated with an octave-band filter bank, but without any implied constraints on the underlying basis functions. This definition is somewhat more relaxed than the formal mathematical definition of wavelet systems (which, for example, imposes constraints on the smoothness of the underlying basis functions). Since our work is equally applicable regardless of whether the looser or more formal definition is employed, we shall not dwell on this matter further.

IV. Symmetric Extension

In the case of conventional wavelet transforms, a popular technique for handing finite-length signals in a nonexpansive manner is symmetric extension. The use of symmetric extension was first proposed by Smith and Eddins [12], and subsequently received considerable attention in the literature (e.g., [11, 18–24]). With this technique, a signal is extended so that it is both symmetric and periodic. Then, both the symmetry and periodicity properties are exploited in order to obtain a transform that is nonexpansive.

Consider the UMD filter bank shown in Fig. 2 which computes a conventional wavelet transform. Suppose that we have a finite-length signal $\hat{x}[n]$ defined for $n = 0, 1, \ldots, N-1$. We then choose $x[n]$, the input to the analysis filter bank, as a symmetric extension of $\hat{x}[n]$. Thus, the period of $x[n]$ is (approximately)

$2N$. In order to be able to handle finite-length signals in a nonexpansive manner, symmetric extension must be employed with a transform that preserves signal symmetry. That is, the transform must be such that an appropriately chosen symmetric input signal $x[n]$ yields symmetric/antisymmetric subband signals $\{y_k[n]\}_{k=0}^{1}$. Therefore, each of the subband signals $\{y_k[n]\}_{k=0}^{1}$ is symmetric/antisymmetric with a period of (approximately) N. Since each of the subband signals $\{y_k[n]\}_{k=0}^{1}$ has symmetry, only (about) one half of the samples of a single period are required to characterize the signal. In other words, we only need (about) $\frac{N}{2}$ samples from each subband signal, for a total of $\frac{N}{2}+\frac{N}{2}=N$ samples. Thus, a nonexpansive transform is obtained.

As indicated above, symmetric extension requires a transform that preserves signal symmetry. Not all conventional wavelet transforms satisfy this constraint. More importantly, even when a conventional wavelet transform does preserve signal symmetry, a reversible ITI version of the same transform may not (due to the addition of rounding operations). Therefore, applying the symmetric extension method in the reversible ITI case is inherently more complicated. Fortunately, as we explain in this manuscript, it is possible to construct reversible ITI wavelet transforms that are compatible with symmetric extension (i.e., satisfy the above symmetry preservation constraint). In fact, in the next section, we present two families of such transforms.

V. Transform Families

In this manuscript, we consider two families of reversible ITI wavelet transforms that are compatible with symmetric extension. Both are derived from the lifting-based parameterizations of linear-phase filter banks presented in [25], and have the general form shown previously in Fig. 1.

Odd-Length Analysis/Synthesis Filter (OLASF) Family

The first family of reversible ITI wavelet transforms is associated with a linear-phase filter bank having odd-length analysis/synthesis filters, and hence is referred to as the odd-length analysis/synthesis filter (OLASF) family. For this family, the parameters of Fig. 1 are constrained as follows:

$$A_k(z) = \begin{cases} \sum_{i=0}^{(L_k-2)/2} a_{k,i}(z^{-i}+z^{i+1}) & \text{for even } k \\ \sum_{i=0}^{(L_k-2)/2} a_{k,i}(z^{-i-1}+z^{i}) & \text{for odd } k, \end{cases} \tag{6}$$

for $k=0,1,\ldots,2\lambda-1$, where the $\{L_k\}_{k=0}^{2\lambda-1}$ are all even integers. The $\{Q_k\}_{k=0}^{2\lambda-1}$ can be chosen as any arbitrary rounding operators. Without loss of generality, we assume that none of the $\{A_k(z)\}_{k=1}^{2\lambda-2}$ are identically zero (i.e., only $A_0(z)$ and $A_{2\lambda-1}(z)$ may be identically zero). In passing, we note that the "odd-length" qualifier in the name "OLASF" refers to the length of the corresponding analysis/synthesis filters (i.e., the $\{H_k\}$ and $\{F_k\}$ in Fig. 2), not the lifting filters (i.e., the $\{A_k\}$ above) which have even length.

One can show that any perfect-reconstruction (PR) linear-phase FIR filter bank with odd-length analysis/synthesis filters has a corresponding OLASF parameterization, provided that the analysis and synthesis filters are normalized appropriately (via a scaling and shift of their impulse responses). The proof is constructive by a matrix Euclidean algorithm (e.g., by using an algorithm based on that described in [16]). Some well known members of the OLASF family that are useful for signal coding include the 5/3, 9/7-M, 5/11-A, 5/11-C, 13/7-T, 13/7-C, and 9/7-F transforms presented in [8].

Even-Length Analysis/Synthesis Filter (ELASF) Family

The second family of reversible ITI wavelet transforms is associated with a linear-phase filter bank having even-length analysis/synthesis filters, and hence is referred to as the even-length analysis/synthesis filter (ELASF) family. For this family, the parameters of Fig. 1 are constrained as follows:

$$A_k(z) = \begin{cases} -1 & \text{for } k = 0 \\ \frac{1}{2} + \hat{A}_1(z) & \text{for } k = 1 \\ \hat{A}_k(z) & \text{for } k \geq 2, \end{cases} \quad \text{and} \tag{7a}$$

$$Q_k \text{ is } \begin{cases} \text{integer-bias invariant} & \text{for k = 1} \\ \text{odd} & \text{for } k \geq 2 \text{ and } k \text{ even} \\ \text{arbitrary} & \text{for } k \geq 2 \text{ and } k \text{ odd,} \end{cases} \tag{7b}$$

for $k = 0, 1, \ldots, 2\lambda - 1$, where

$$\hat{A}_k(z) = \sum_{i=1}^{(L_k-1)/2} \hat{a}_{k,i}(z^{-i} - z^i), \tag{7c}$$

$\lambda \geq 1$, and the $\{L_k\}_{k=1}^{2\lambda-1}$ are all odd integers. (The operator Q_0 is simply chosen as the identity, since the output of the filter A_0 is always an integer.) Without loss of generality, we assume that none of the $\{\hat{A}_k(z)\}_{k=2}^{2\lambda-2}$ are identically zero (i.e., only $\hat{A}_1(z)$ and $\hat{A}_{2\lambda-1}(z)$ may be identically zero). In passing, we note that the "even-length" qualifier in the name "ELASF" refers to the length of the corresponding analysis/synthesis filters (i.e., the $\{H_k\}$ and $\{F_k\}$ in Fig. 2), not the lifting filters (i.e., the $\{A_k\}$ above) which have odd length.

The ELASF transform family is a generalization of ones commonly considered in the literature (e.g., [26]). Our proposed family is a generalization in two respects. First, in the case of transform families with a similar structure, it is often suggested that the $\{Q_k\}_{k=2}^{2\lambda-1}$ must all be chosen as odd functions. Such a choice, however, is overly restrictive. The rounding operators $\{Q_k\}_{k=2}^{2\lambda-1}$ only need to be chosen as odd

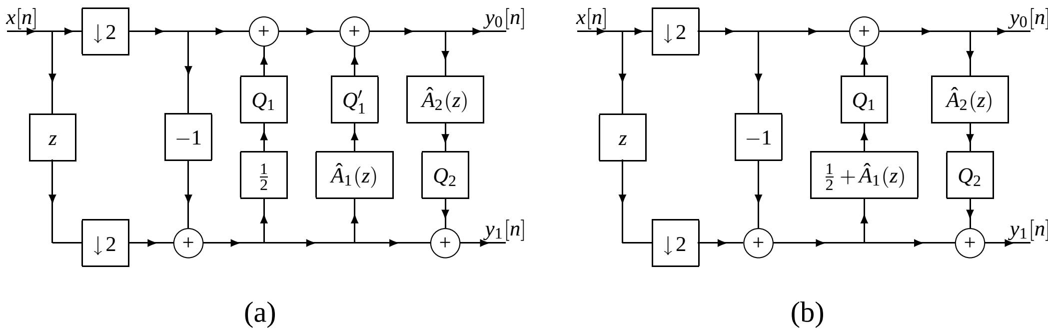

Fig. 3. Two different reversible ITI approximations for the 6/14 transform. The analysis filter bank based on the (a) commonly referenced parameterization and (b) our more general parameterization. $\hat{A}_1(z) = -\frac{1}{16}(z - z^{-1})$, $\hat{A}_2(z) = \frac{1}{16}\left((z^2 - z^{-2}) - 6(z - z^{-1})\right)$, Q_1 is integer-bias invariant, Q'_1 is arbitrary, and Q_2 is odd.

functions for even k. Second, and more fundamentally, the related families commonly considered in the literature (e.g., as in [26]) are essentially a special case of our proposed family with $\hat{A}_1(z) \equiv 0$.

The additional flexibility afforded by the introduction of the $\hat{A}_1(z)$ parameter has important practical implications. In many cases, by using our more general parameterization, the number of lifting steps can be reduced by one. By reducing the number of lifting steps, the number of rounding operations is also decreased, leading to a reduction in rounding error. Consider, for example, the two ITI filter banks shown in Fig. 3. One can easily verify that both filter banks approximate the same linear transform, namely the one associated with the 6/14 transform [8] which is known to be effective for image coding purposes. Both filter banks are reversible, as both employ lifting/ladder networks for polyphase filtering, but due to the nonlinearities introduced by rounding, the two filter banks are not equivalent. Comparing the filter banks of Figs. 3(a) and 3(b), we can see that the former has one more rounding operation than the latter. For this reason, the filter bank in Fig. 3(b) will normally have better approximation properties (i.e., smaller rounding error) than the filter bank of Fig. 3(a). Although not immediately obvious, like the filter bank in Fig. 3(a), the filter bank in Fig. 3(b) is also compatible with the symmetric extension method, as we will show in a later section.

Only a subset of all transforms associated with PR linear-phase FIR filter banks with even-length analysis/synthesis filters belong to the ELASF family. Moreover, this subset is quite small. In this work, we consider only those transforms belonging to the ELASF family because our interest lies with transforms that are compatible with symmetric extension. As we will demonstrate later, transforms in the ELASF family are compatible with this method. It is not clear, however, how (or even if) the compatibility issue can be handled in the more general even-length analysis/synthesis filter case. (For example, the associated matrix factorization problem has been studied in [27], but the scheme presented therein does not generally yield transforms that are compatible with symmetric extension.) Notwithstanding the constrained nature of the ELASF family, it still contains many practically useful transforms. Some well known members of this family that are useful for signal coding include (minor variations on) the 2/6, 2/10, and 6/14 transforms discussed in [8] and the S transform [6].

Derivation of the Transform Families

In essence, the preceding families were derived by observing that if the forward polyphase transform (i.e., shifting followed by downsampling) preserves symmetry and the subsequent lifting steps collectively preserve symmetry, then the overall transform will also preserve symmetry and therefore be compatible with symmetric extension. Fortunately, the forward polyphase transform does not pose much of a problem, as it is relatively easy to have this process preserve symmetry. The difficulty is in finding a general set of lifting steps that collectively preserve symmetry. This is most easily achieved by requiring that all/most of the lifting steps individually preserve symmetry. In turn, this is accomplished by choosing all/most of the lifting step filters to have linear phase. The specifics of why this approach works will become clear in the next section.

VI. Compatibility of Transform Families with Symmetric Extension

In the previous section, we introduced the OLASF and ELASF families of reversible ITI wavelet transforms. Now, we show that the transforms from these two families are compatible with symmetric extension (i.e., can be used with symmetric extension in order to handle finite-length signals in a nonexpansive manner). Consider a filter bank of the form shown in Fig. 1 that is constrained to be of the OLASF type as defined by (6) or ELASF type as defined by (7). Suppose that we are given a signal $\hat{x}[n]$ defined for $n = 0, 1, \ldots, N-1$. By using symmetric extension, one can form a nonexpansive transform, as demonstrated by the two propositions and accompanying proofs below.

Proposition 1 (Compatibility of the OLASF family with symmetric extension) Consider a filter bank of the form shown in Fig. 1 that is constrained to be of the OLASF type as defined by (6). Suppose now that we are given a signal $\hat{x}[n]$ defined for $n = 0, 1, \ldots, N-1$. If we choose $x[n]$, the input to the analysis filter bank, as the symmetric extension of $\hat{x}[n]$ given by

$$x[n] = \hat{x}[\min(\operatorname{mod}(n-K, 2N-2), 2N-2-\operatorname{mod}(n-K, 2N-2))], \quad K \in \mathbb{Z} \tag{8}$$

(i.e., $x[n]$ is defined such that $x[n] = x[n+2N-2]$ and $x[n] = x[2K-n]$ for all $n \in \mathbb{Z}$), then 1) $y_0[n]$ is completely characterized by its N_0 samples at indices $\lceil \frac{K}{2} \rceil, \lceil \frac{K}{2} \rceil + 1, \ldots, \lfloor \frac{K+N-1}{2} \rfloor$; 2) $y_1[n]$ is completely characterized by its N_1 samples at indices $\lceil \frac{K-1}{2} \rceil, \lceil \frac{K-1}{2} \rceil + 1, \ldots, \lfloor \frac{K+N-2}{2} \rfloor$; and 3) $N_0 + N_1 = N$ (i.e., the transform is nonexpansive).

Proof: Consider the OLASF-type system associated with Fig. 1. Evidently, $x[n]$ is $(2N-2)$-periodic and symmetric about K and $K+N-1$ (with additional pairs of symmetry points following from periodicity). From the properties of downsampling, one can show that $u_0[n]$ is $(N-1)$-periodic and symmetric about $\frac{K}{2}$ and $\frac{K+N-1}{2}$, and $v_0[n]$ is $(N-1)$-periodic and symmetric about $\frac{K-1}{2}$ and $\frac{K+N-2}{2}$. Consider now the first lifting step (associated with filter A_0). Since, from (6), the filter A_0 has a group delay of $-\frac{1}{2}$ and the rounding operator Q_0 preserves signal symmetry, the adder (at the output of Q_0) is summing two $(N-1)$-periodic symmetric signals with the same symmetry centers. Thus, the adder output, $v_1[n]$, is also $(N-1)$-periodic and symmetric with the same symmetry centers, namely $\frac{K-1}{2}$ and $\frac{K+N-2}{2}$. Consider now the second lifting step (associated with filter A_1). Since, from (6), the filter A_1 has a group delay of $\frac{1}{2}$ and the rounding operator Q_1 preserves signal symmetry, both adder inputs are $(N-1)$-periodic symmetric signals with the same symmetry centers. Therefore, the adder output, $u_2[n]$, must also be $(N-1)$-periodic and symmetric with the same symmetry centers, namely $\frac{K}{2}$ and $\frac{K+N-1}{2}$. By repeating the above reasoning for each of the remaining lifting steps, we have that the $\{u_k[n]\}_{k=0}^{2\lambda-1}$ and $y_0[n]$ are all $(N-1)$-periodic and symmetric about $\frac{K}{2}$ and $\frac{K+N-1}{2}$. Likewise, the $\{v_k[n]\}_{k=0}^{2\lambda-2}$ and $y_1[n]$ are all $(N-1)$-periodic and symmetric about $\frac{K-1}{2}$ and $\frac{K+N-2}{2}$.

By examining the form of the various signals, we can see that the $\{u_k[n]\}_{k=0}^{2\lambda-1}$ and $y_0[n]$ are completely characterized by the N_0 samples at indices $\lceil\frac{K}{2}\rceil, \lceil\frac{K}{2}\rceil+1, \ldots, \lfloor\frac{K+N-1}{2}\rfloor$, and the $\{v_k[n]\}_{k=0}^{2\lambda-2}$ and $y_1[n]$ are completely characterized by the N_1 samples at indices $\lceil\frac{K-1}{2}\rceil, \lceil\frac{K-1}{2}\rceil+1, \ldots, \lfloor\frac{K+N-2}{2}\rfloor$, where

$$N_0 \triangleq \left\lfloor\frac{K+N-1}{2}\right\rfloor - \left\lceil\frac{K}{2}\right\rceil + 1 \quad \text{and} \tag{9}$$

$$N_1 \triangleq \left\lfloor\frac{K+N-2}{2}\right\rfloor - \left\lceil\frac{K-1}{2}\right\rceil + 1. \tag{10}$$

Using (5), we can further simplify (9) and (10) to obtain

$$N_0 = \begin{cases} \frac{N}{2} & \text{for even } N \\ \frac{N+1}{2} & \text{for odd } N\text{, even } K \\ \frac{N-1}{2} & \text{for odd } N\text{, odd } K, \end{cases} \quad \text{and} \quad N_1 = \begin{cases} \frac{N}{2} & \text{for even } N \\ \frac{N-1}{2} & \text{for odd } N\text{, even } K \\ \frac{N+1}{2} & \text{for odd } N\text{, odd } K. \end{cases} \tag{11}$$

Thus, from (11), we have that, regardless of the parities of K and N, $N_0 + N_1 = N$. In other words, the total number of independent samples in the transformed signal is always equal to the number of samples in the original signal, and the resulting transform is, therefore, nonexpansive. ■

Proposition 2 (Compatibility of the ELASF family with symmetric extension) Consider a filter bank of the form shown in Fig. 1 that is constrained to be of the ELASF type as defined by (7). Suppose

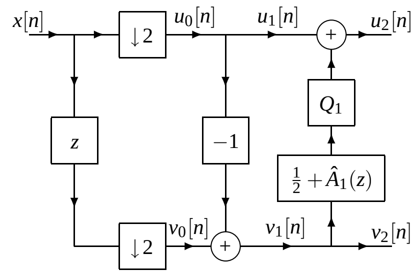

Fig. 4. Base analysis filter bank for the ELASF family of transforms.

now that we are given a signal $\hat{x}[n]$ defined for $n = 0, 1, \ldots, N-1$. If we choose $x[n]$, the input to the analysis filter bank, as the symmetric extension of $\hat{x}[n]$ given by

$$x[n] = \hat{x}[\min(\operatorname{mod}(n-K, 2N), 2N-1-\operatorname{mod}(n-K, 2N))], \quad K \in \mathbb{Z} \tag{12}$$

(i.e., $x[n]$ is defined such that $x[n] = x[n+2N]$ and $x[n] = x[2K-1-n]$ for all $n \in \mathbb{Z}$), then 1) $y_0[n]$ is completely characterized by its N_0 samples at indices $\lceil \frac{K-1}{2} \rceil, \lceil \frac{K-1}{2} \rceil + 1, \ldots, \lfloor \frac{K+N-1}{2} \rfloor$; 2) $y_1[n]$ is completely characterized by its N_1 samples at indices $\lceil \frac{K}{2} \rceil, \lceil \frac{K}{2} \rceil + 1, \ldots, \lfloor \frac{K+N-2}{2} \rfloor$; and 3) $N_0 + N_1 = N$ (i.e., the transform is nonexpansive).

Proof: Consider a transform in the ELASF family. Such a transform is associated with the filter bank shown in Fig. 1 having the corresponding parameters given by (7). The base part of the analysis filter bank has the form shown in Fig. 4. Now, let us consider the linear version of this filter bank as depicted in Fig. 5. This filter bank can be represented in canonical form as shown in Fig. 2, where the transfer functions of the lowpass and highpass analysis filters are denoted as $H_0(z)$ and $H_1(z)$, respectively. Using the noble identities [28], we can show that

$$H_0(z) = \tfrac{1}{2}(z+1) + (z-1)\hat{A}_1(z^2) \quad \text{and} \quad H_1(z) = z-1. \tag{13}$$

Due to the form of $\hat{A}_1(z)$ in (7), it follows from (13) that the filter H_0 has an impulse response symmetric about $-\frac{1}{2}$. From (13), the filter H_1 has an impulse response that is antisymmetric about $-\frac{1}{2}$. Consequently, in Fig. 5, if $x[n]$ is generated using (12), we have that $u_2[n]$ is N-periodic and symmetric about $\frac{K-1}{2}$ and $v_2[n]$ is N-periodic and antisymmetric with the same symmetry center as $u_2[n]$. Since $u_2[n]$ and $v_2[n]$ are N-periodic, it follows that both signals must also have another symmetry center at $\frac{K+N-1}{2}$.

Now, suppose that we round the lowpass subband output $u_2[n]$ in Fig. 5, so as to obtain the system shown in Fig. 6. For any (reasonable) rounding function, this process will maintain signal symmetry. Moreover, if the rounding operator Q is integer-bias invariant, we can equivalently move the rounding operator to the input side of the adder, resulting in the system in Fig. 4 (where $Q_1 = Q$). Consequently, in the case of the system in Fig. 4, if the input $x[n]$ has the form of (12), $u_2[n]$ must be N-periodic

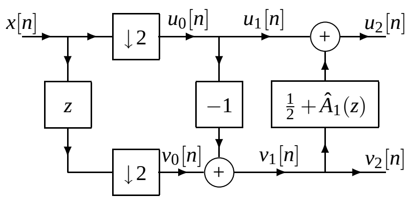

Fig. 5. Linear version of the base analysis filter bank for the ELASF family of transforms.

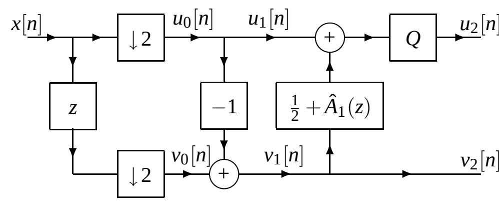

Fig. 6. Modified base analysis filter bank for the ELASF family of transforms.

and symmetric about $\frac{K-1}{2}$ and $\frac{K+N-1}{2}$, while $v_2[n]$ must be N-periodic and antisymmetric with the same symmetry centers as $u_2[n]$.

Now, let us return to the system in Fig. 1. From above, we have that $u_2[n]$ is N-periodic and symmetric about $\frac{K-1}{2}$ and $\frac{K+N-1}{2}$, and $v_2[n]$ is N-periodic and antisymmetric with the same symmetry points as $u_2[n]$. Consider now the third lifting step (associated with filter A_2). Since, from (7), the filter A_2 has a group delay of zero, and the rounding operator Q_2 preserves signal antisymmetry (since it is odd), the adder (at the output of Q_2) is summing two N-periodic antisymmetric signals with the same symmetry centers. Thus, the adder output is also N-periodic and antisymmetric with the same symmetry centers. Consider the fourth lifting step (associated with filter A_3). Since, from (7), the filter A_3 has a group delay of zero, and the rounding operator Q_3 preserves signal symmetry, the adder is summing two N-periodic symmetric signals with the same symmetry centers. Thus, the adder output is also N-periodic and symmetric with the same symmetry centers. It is important to note that, for odd k, Q_k need not be odd, contrary to the suggestions of some, in the case of similar parameterizations. By repeating the above reasoning for each of the remaining lifting steps, we have that the $\{u_k[n]\}_{k=2}^{2\lambda-1}$ and $y_0[n]$ are N-periodic and symmetric with symmetry centers $\frac{K-1}{2}$ and $\frac{K+N-1}{2}$, and the $\{v_k[n]\}_{i=1}^{2\lambda-2}$ and $y_1[n]$ are N-periodic and antisymmetric with the same symmetry centers.

By examining the form of the various signals, we note that the $\{u_k[n]\}_{k=2}^{2\lambda-1}$ and $y_0[n]$ are completely characterized by the N_0 samples at indices $\lceil\frac{K-1}{2}\rceil, \lceil\frac{K-1}{2}\rceil+1, \ldots, \lfloor\frac{K+N-1}{2}\rfloor$, and the $\{v_k[n]\}_{k=1}^{2\lambda-2}$ and $y_1[n]$ are completely characterized by the N_1 samples at indices $\lceil\frac{K}{2}\rceil, \lceil\frac{K}{2}\rceil+1, \ldots, \lfloor\frac{K+N-2}{2}\rfloor$, where

$$N_0 \triangleq \lfloor\tfrac{K+N-1}{2}\rfloor - \lceil\tfrac{K-1}{2}\rceil + 1 \quad \text{and} \tag{14}$$

$$N_1 \triangleq \lfloor\tfrac{K+N-2}{2}\rfloor - \lceil\tfrac{K}{2}\rceil + 1. \tag{15}$$

Using (5), we can further simplify (14) and (15) to obtain

$$N_0 = \begin{cases} \frac{N+1}{2} & \text{for } N \text{ odd} \\ \frac{N}{2} & \text{for } N \text{ even, } K \text{ even} \\ \frac{N+2}{2} & \text{for } N \text{ even, } K \text{ odd,} \end{cases} \quad \text{and} \quad N_1 = \begin{cases} \frac{N-1}{2} & \text{for } N \text{ odd} \\ \frac{N}{2} & \text{for } N \text{ even, } K \text{ even} \\ \frac{N-2}{2} & \text{for } N \text{ odd, } K \text{ odd.} \end{cases} \tag{16}$$

Thus, from (16), we have that, regardless of the parities of K and N, $N_0 + N_1 = N$. In other words, the total number of independent samples in the transformed signal is always equal to the number of samples in the original signal, and the resulting transform is, therefore, nonexpansive. ■

In the above propositions, the parameter K serves to introduce a shift in the input signal $x[n]$. Since downsampling is only periodically shift invariant (with a period of two), two different modes of behavior are possible depending on the parity (i.e., evenness/oddness) of K. For both the OLASF and ELASF families of transforms, it can sometimes be convenient, in practice, to choose the parameter K as zero. This forces the first characteristic/independent sample for both subband signals to be located at an index of zero, which may be convenient for bookkeeping purposes. In some applications, however, an arbitrary (or nonzero) value for K may be desirable—hence, our motivation for considering the more general case here.

VII. ELASF Family: A Closer Look

Earlier, we introduced the ELASF family of transforms. In this section, we explore some of the characteristics of transforms in this family. In particular, we consider the properties of the linear systems associated with the ELASF family. Such results provide new insight into the transforms of the ELASF family.

Consider the linear version of a filter bank of the form shown in Fig. 1(a) that is constrained to be of the ELASF type as defined by (7). (By "linear version", we mean the filter bank obtained by simply setting all of the $\{Q_k\}_{k=0}^{2\lambda-1}$ equal to the identity.) Let us represent the analysis filter bank in its canonical form (as shown in Fig. 2(a)), where the transfer functions of the lowpass analysis, highpass analysis, lowpass synthesis, and highpass synthesis filters are denoted as $H_0(z)$, $H_1(z)$, $F_0(z)$, and $F_1(z)$, respectively. Further, we define the transfer matrix for part of the analysis polyphase filtering network as follows:

$$\begin{aligned} \mathbf{P}(z) &\stackrel{\triangle}{=} \begin{bmatrix} P_{0,0}(z) & P_{0,1}(z) \\ P_{1,0}(z) & P_{1,1}(z) \end{bmatrix} \\ &= \left(\prod_{k=1}^{\lambda-1} \begin{bmatrix} 1 & \hat{A}_{2k+1}(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \hat{A}_{2k}(z) & 1 \end{bmatrix} \right) \begin{bmatrix} 1 & \hat{A}_1(z) \\ 0 & 1 \end{bmatrix}. \end{aligned} \tag{17}$$

Using the noble identities [28] and straightforward algebraic manipulation, we can show

$$\begin{bmatrix} H_0(z) \\ H_1(z) \end{bmatrix} = \mathbf{P}(z^2) \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ z \end{bmatrix} = \mathbf{P}(z^2) \begin{bmatrix} \frac{1}{2}(z+1) \\ z-1 \end{bmatrix}.$$

For convenience, can rewrite the preceding equation as follows:

$$H_0(z) = \tfrac{1}{2}(z+1)P_{0,0}(z^2) + (z-1)P_{0,1}(z^2) \quad \text{and} \tag{18a}$$

$$H_1(z) = (z-1)P_{1,1}(z^2) + \tfrac{1}{2}(z+1)P_{1,0}(z^2). \tag{18b}$$

A detailed analysis of (17) reveals that $\mathbf{P}(z)$ has a number of important properties as given by the proposition in Appendix I. Using this proposition, we can show that: 1) $\mathcal{Z}^{-1}H_0(z)$ and $\mathcal{Z}^{-1}H_1(z)$ both have symmetry about $-\frac{1}{2}$ and begin and end with pairs of coefficients that are either equal or equal in magnitude but opposite in sign; 2) H_0 and H_1 are always of even length, and cannot have the same length except in the degenerate case where all of the $\{\hat{A}_k(z)\}_{k=1}^{2\lambda-1}$ are identically zero; 3) the DC and Nyquist gains of H_0 and H_1 are fixed and given by $|H_0(1)| = 1$, $|H_0(-1)| = 0$, $|H_1(-1)| = 2$, and $|H_1(1)| = 0$; and 4) by swapping the analysis and synthesis filters (and renormalizing), one can always obtain a new filter bank that is associated with a transform in the ELASF family[1]. A proof of the preceding four properties can be found in Appendix II.

In what follows, we explain the significance of the above four properties. The first two properties provide some insight into why the ELASF family only generates a subset of filter banks with even-length analysis/synthesis filters. For example, from the second property, we can see that the ELASF family does not include transforms associated with equal-length lowpass and highpass analysis (or synthesis) filters. Furthermore, the first property implies that even in the unequal-length case, the ELASF family still does not constitute a complete parameterization (e.g., due to the close relationship between the first/last two samples in $\mathcal{Z}^{-1}H_0(z)$ and also $\mathcal{Z}^{-1}H_1(z)$). The third property is of practical interest, since it is often desirable for a reversible ITI wavelet transform to have a corresponding lowpass analysis filter with a DC gain of one (which typically results in a transform with good dynamic range properties [8]). Moreover, this property also demonstrates that the lowpass and highpass analysis filters always have reasonable frequency responses (i.e., zeros at the Nyquist and DC frequencies, respectively). The fourth property is also practically useful, since, in some cases, the "transposed" filter bank (i.e., the one with the analysis and synthesis filters swapped) may also be effective for signal coding purposes.

VIII. New Symmetry-Preserving Reversible ITI Structures

As noted earlier, the preservation of signal symmetry is of critical importance when symmetric extension is to be employed. In this section, we further examine the problem of maintaining signal symmetry in reversible ITI wavelet transforms.

[1]To date, this fact seems to have been overlooked (e.g., as in [26]).

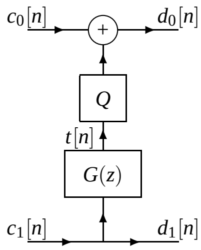

Fig. 7. Lifting step.

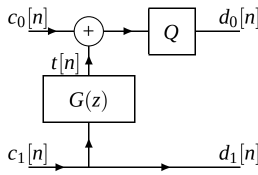

Fig. 8. Modified lifting step.

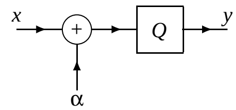

Fig. 9. Network consisting of an adder and rounding unit.

To begin, we consider one of the basic building blocks of such transforms, the lifting step. A lifting step normally has the form shown in Fig. 7 and is invertible for any choice of Q. From the standpoint of symmetry preservation, however, it can be advantageous to apply the rounding operator Q after the adder as in Fig. 8. Unfortunately, the systems in Figs. 7 and 8 are only equivalent if Q is integer-bias invariant. In certain circumstances, the choice of an odd Q is more appropriate (which implies Q is not integer-bias invariant [13], as discussed before). In particular, such a choice is beneficial when dealing with antisymmetric signals in the context of symmetric extension. Consider what happens in the systems of both Figs. 7 and 8 when Q is odd and neither $t[n]$ nor $c_0[n]$ have any particular symmetry but $t[n]+c_0[n]$ is known to be antisymmetric. In the former case (i.e., Fig. 7), $d_0[n]$ will not normally be antisymmetric. That is, antisymmetry is not preserved (by rounding). In the latter case (i.e., Fig. 8), however, $d_0[n]$ is guaranteed to be antisymmetric (i.e., signal antisymmetry is preserved).

From the above discussion, we can see that using a modified lifting step as in Fig. 8 can sometimes be advantageous. There is, however, one problem with such an approach. If Q is not integer-bias invariant, it is not clear that the structure in Fig. 8 has an inverse. In what follows, we will examine this question of invertibility more closely.

Consider again a network of the form shown in Fig. 8. Obviously, this network is invertible if we can invert the network in Fig. 9, where $x, y \in \mathbb{Z}$ and $\alpha \in \mathbb{R}$. That is, mathematically, the network in Fig. 8 is invertible if we can invert (i.e., solve uniquely for x in terms of y) the equation $y = Q(\alpha + x)$, where $x, y \in \mathbb{Z}$ and α is a real constant. We assert that this equation has an inverse if Q is chosen as either the biased truncation or RAFZ function, but not if Q is chosen as the truncation function. This assertion is shown to be correct by the proposition and accompanying proof below.

Proposition 3: Consider the equations

$$y = \operatorname{trunc}(\alpha + x), \tag{19}$$

$$y = \operatorname{btrunc}(\alpha + x), \quad \text{and} \tag{20}$$

$$y = \operatorname{rafz}(\alpha + x), \tag{21}$$

where $x, y \in \mathbb{Z}$ and α is a real constant. If $\alpha \notin \mathbb{Z}$, (19) is not invertible (i.e., one cannot always uniquely solve for x in terms of y). For any $\alpha \in \mathbb{R}$, (20) and (21) are invertible, with their respective inverses given by

$$x = \begin{cases} y - \alpha - \frac{1}{2}\operatorname{sgn} y & \text{if } \alpha \text{ is an odd integer multiple of } \frac{1}{2} \\ y - \operatorname{bfloor} \alpha & \text{otherwise,} \end{cases} \quad \text{and} \tag{22}$$

$$x = \begin{cases} y - \lceil \alpha \rceil & \text{for } y \geq 0 \\ y - \lfloor \alpha \rfloor & \text{for } y < 0. \end{cases} \tag{23}$$

Proof: **First, we show that (19) is not invertible:** Since $\alpha \notin \mathbb{Z}$, from the definition of the floor function, we have $0 < \alpha - \lfloor \alpha \rfloor < 1$ which also implies $-1 < \alpha - \lfloor \alpha \rfloor - 1 < 0$. In turn, the preceding two relationships imply, respectively, that $\operatorname{trunc}(\alpha - \lfloor \alpha \rfloor) = 0$ and $\operatorname{trunc}(\alpha - \lfloor \alpha \rfloor - 1) = 0$. Thus, for any $\alpha \notin \mathbb{Z}$, we have that $\operatorname{trunc}(\alpha + x) = 0$ for both $x = -\lfloor \alpha \rfloor$ and $x = -\lfloor \alpha \rfloor - 1$. Since two distinct values for x both yield the same value for y, (19) cannot be invertible.

Next, we show that (20) is inverted by (22): Let us begin by considering the case that α is an odd integer multiple of $\frac{1}{2}$. From the definition of the btrunc function in (1), we can rewrite (20) as

$$y = \begin{cases} \lfloor \alpha + x + \frac{1}{2} \rfloor & \text{for } \alpha + x \geq 0 \\ \lceil \alpha + x - \frac{1}{2} \rceil & \text{for } \alpha + x < 0. \end{cases}$$

Since α is an odd integer multiple of $\frac{1}{2}$, we know that $\alpha + x + \frac{1}{2}$ and $\alpha + x - \frac{1}{2}$ are both integers, and we can rewrite the preceding equation as $y = \alpha + x + \frac{1}{2}\operatorname{sgn}(\alpha + x)$. From (20), we can deduce that $\operatorname{sgn}(\alpha + x) = \operatorname{sgn} y$, since $\operatorname{sgn}(\operatorname{btrunc}(\alpha + x)) = \operatorname{sgn}(\alpha + x)$ if $|\alpha + x| \geq \frac{1}{2}$ which must be the case here. Consequently, we can solve for x to obtain $x = y - \alpha - \frac{1}{2}\operatorname{sgn} y$. Thus, we have proven (22) to be correct in the case that α is an odd integer multiple of $\frac{1}{2}$.

Now let us consider the case that α is not an odd integer multiple of $\frac{1}{2}$. Using (1) and (4), we can rewrite (20) as $y = \operatorname{bfloor}(\alpha + x) = x + \operatorname{bfloor} \alpha$. Solving for x, we obtain $x = y - \operatorname{bfloor} \alpha$. Thus, we have proven (22) to be correct in the case that α is not an odd integer multiple of $\frac{1}{2}$. Therefore, (20) is inverted by (22).

Lastly, we show that (21) is inverted by (23): Using the definition of the RAFZ function in (2), and solving for x, we obtain

$$x = \begin{cases} y - \lceil \alpha \rceil & \text{for } x + \alpha \geq 0 \\ y - \lfloor \alpha \rfloor & \text{for } x + \alpha < 0. \end{cases} \tag{24}$$

From the definition of the RAFZ function, however, we know that $\operatorname{sgn}(\operatorname{rafz} \alpha) = \operatorname{sgn} \alpha$ for all $\alpha \in \mathbb{R}$ (i.e., the RAFZ function preserves signedness). Consequently, we have that $\operatorname{sgn}(x+\alpha) = \operatorname{sgn} y$, and (24) can be simplified to obtain (23). Thus, (21) is inverted by (23). ■

From the above proposition, we consequently have that the modified lifting step in Fig. 8 can be inverted if Q is chosen as the biased truncation or RAFZ function, but not if Q is chosen as the truncation function. We can exploit this knowledge in order to gain increased flexibility in the construction of symmetry-preserving reversible ITI networks.

For example, consider the linear-phase analysis filter bank shown in Fig. 10, where $\hat{A}_1(z)$ is of the form given in (7c). Due to the linear-phase property of the analysis filters, this linear filter bank is compatible with symmetric extension. In particular, one can show that if $x[n]$ is generated using (12), then $y_0[n]$ is symmetric and $y_1[n]$ is antisymmetric. We can, therefore, obtain a symmetry-preserving ITI version of this filter bank by simply rounding the output of the highpass subband signal using an odd rounding function (since signal antisymmetry must be preserved). In so doing, we obtain the new structure shown in Fig. 11, where Q is a rounding operator. Since Q is odd, however, Q cannot be integer-bias invariant. Consequently, we cannot equivalently move the rounding unit before the adder. We know from our previous results, however, that if Q is chosen as either the biased truncation or RAFZ function, the resulting filter bank (in Fig. 11) must be invertible. Furthermore, in each case, the corresponding inverse filter bank can be easily deduced by using Proposition 3 (from above). Thus, we have, in effect, constructed a new symmetry-preserving base filter bank, similar to (but distinct from) the one used in the ELASF family. Moreover, the newly proposed reversible ITI structures may also prove useful for the construction of other more general symmetry-preserving reversible ITI transforms.

IX. Relationship Between Symmetric Extension and Per-Lifting-Step Extension

Although symmetric extension is quite popular, other extension schemes can also be devised. For example, one extension technique which can be used in conjunction with lifting-based transforms is the per-lifting-step (PLS) extension method. We have mentioned this technique briefly in [8] and used it in the SBTLIB [29] and JasPer [30, 31] software. More recently, this method has also been described in [26, 32] (under the name of “iterated extension” or “interleaved extension”).

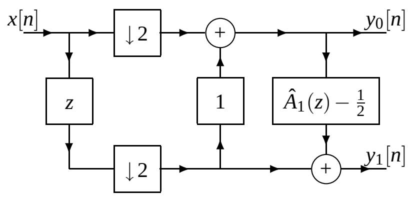

Fig. 10. Linear version of the base analysis filter bank for the new transform family.

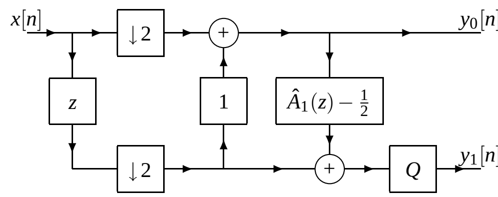

Fig. 11. Modified base analysis filter bank for the new transform family.

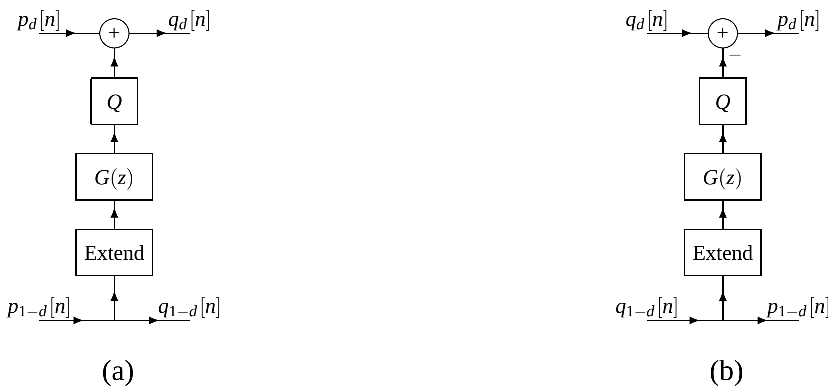

Fig. 12. Structure of a lifting step in the case of PLS extension. (a) Forward lifting step and (b) inverse lifting step.

With PLS extension, signal extension is performed at the input to each lifting-step filter rather than being performed at the input to the filter bank. That is, each lifting step on the analysis side of the filter bank has the form shown in Fig. 12(a). In the diagram, $p_0[n]$ and $q_0[n]$ represent intermediate lowpass channel signals, $p_1[n]$ and $q_1[n]$ represent intermediate highpass channel signals, and Q is a rounding operator. On the synthesis side of the filter bank, the lifting step of Fig. 12(a) has a corresponding inverse of the form shown in Fig. 12(b). Clearly, the same extension values can be generated on both the analysis and synthesis sides (since $p_{1-d}[n] = q_{1-d}[n]$). Thus, this scheme can generate reversible transforms. For example, one might extend the lifting-step filter input to the left by repeating its leftmost sample and to the right by repeating its rightmost sample. In what follows, we refer to this simple case of PLS extension as constant PLS extension.

Consider again a system of the form shown in Fig. 1. In Section VI, we carefully examined the symmetry properties of the signals $\{u_k[n]\}_{k=0}^{2\lambda-1}$, $\{v_k[n]\}_{k=0}^{2\lambda-2}$, $y_0[n]$, and $y_1[n]$. In the OLASF case, all of these signals are symmetric, while in the ELASF case, all of these signals are symmetric/antisymmetric, except $u_0[n]$, $u_1[n]$, and $v_0[n]$. Therefore, we can equivalently define symmetric extension in terms of PLS extension in which case the input to each lifting-step filter is extended by symmetric extension (with the appropriate choice of symmetry type and centers). This general equivalence is, for example, exploited in the SBTLIB software [29] and also briefly noted in [26]. In what follows, however, we would like to more carefully examine the equivalence in the special case of OLASF transforms with length-2 lifting-step filters.

Suppose that we have a filter bank of the form shown in Fig. 1 that is constrained to be of the OLASF type as defined by (6) with $L_k \le 2$ for $k = 0, 1, \ldots, 2\lambda - 1$. In this case, we assert that symmetric extension (using (8) with $K = 0$) is equivalent to constant PLS extension. In the case of both symmetric extension and constant PLS extension, the signals $\{u_k[n]\}_{k=0}^{2\lambda-1}$ and $y_0[n]$ are completely characterized by their samples at indices $n = 0, 1, \ldots, \lfloor \frac{N-1}{2} \rfloor$ and the signals $\{v_k[n]\}_{k=0}^{2\lambda-2}$ and $y_1[n]$ are completely characterized by their samples at indices $n = 0, 1, \ldots, \lfloor \frac{N-2}{2} \rfloor$. Furthermore, both extension methods yield the same $u_0[n]$ and $v_0[n]$ for n over their characteristic sample indices.

Consider the lifting steps involving the filters $\{A_k\}$ for even k. When filtering with the filter A_k, $u_k[n]$ only ever requires right extension by one sample (if at all) in order to obtain the value for $u_k[\lfloor \frac{N+1}{2} \rfloor]$. Suppose first that N is even. In the case of symmetric extension, the symmetry center of $u_k[n]$ at $\frac{N-1}{2}$ is an odd multiple of $\frac{1}{2}$, so the sample obtained by extension is equal to $u_k[\frac{N-2}{2}]$. This, however, is the same result obtained by constant PLS extension. Finally, if N is odd, one less sample needs to be computed for $v_{k+1}[n]$ than $u_k[n]$, and $u_k[n]$ need not be extended at all. Thus, symmetric extension is equivalent to constant PLS extension for the lifting steps involving the filters $\{A_k\}$ for even k.

Consider the lifting steps involving the filters $\{A_k\}$ for odd k. When filtering with the filter A_k, $v_k[n]$ always requires left extension by one sample in order to obtain the value for $v_k[-1]$. In the case of symmetric extension, since a symmetry center of $v_k[n]$ is $-\frac{1}{2}$, the sample obtained by extension is equal to $v_k[0]$. Clearly, this is the same result obtained by constant PLS extension. If N is odd, $v_k[n]$ must also be right extended by one sample in order to obtain the value for $v_k[\frac{N-1}{2}]$ (since one fewer sample is associated with $v_k[n]$ than $u_{k+1}[n]$). In the case of symmetric extension, the symmetry center $\frac{N-2}{2}$ is an odd multiple of $\frac{1}{2}$, so the sample obtained by extension is equal to $v_k[\frac{N-3}{2}]$. Again, this is the same result that is obtained from constant PLS extension. Thus, symmetric extension is equivalent to constant PLS extension for the lifting steps involving the filters $\{A_k\}$ for odd k.

Combining the above results for both sets of lifting-step filters, we see that constant PLS extension is equivalent to symmetric extension for the specific case considered (i.e., OLASF family, length-2 filters, $K = 0$). Since the implementation of constant PLS extension is simpler than the classic implementation of symmetric extension, this equivalence is potentially quite useful. For example, both of the filter banks defined in the JPEG-2000 Part-1 standard (i.e., [4]) are of the form assumed above. Therefore, one can exploit the equivalence between symmetric extension and constant PLS extension in order to simplify

JPEG-2000 codec implementations. For example, this equivalence has been employed by the JasPer software [30, 31] since at least version 0.044.

X. Conclusions

In this manuscript, we considered symmetric extension as a means for constructing nonexpansive reversible ITI wavelet transforms for finite-length signals. We explained how symmetric extension, commonly used in the case of conventional wavelet transforms, can be applied in the reversible ITI case. Two families of reversible ITI wavelet transforms were introduced (i.e., the OLASF and ELASF families), and the transforms from these families were shown to be compatible with symmetric extension. For the more constrained of the two families (i.e., the ELASF family), we characterized the transforms belonging to this family. That is, we showed that: 1) such transforms are associated with analysis filters having transfer functions of a highly structured form; 2) the DC and Nyquist gains of the analysis filters are fixed (at useful values), independent of the choice of free parameters; and 3) if a particular filter bank is associated with a transform in the ELASF family, then so too is its "transposed" version. By better understanding the characteristics of this transform family, one can hope to better utilize it in signal coding applications. During the course of our work, we also derived some new reversible ITI structures that are useful in conjunction with techniques like symmetric extension. We also examined the relationship between symmetric extension and PLS extension. For OLASF transforms associated with length-2 lifting-step filters, we showed that symmetric extension is equivalent to constant PLS extension. This fact can be exploited in order to reduce the complexity of JPEG-2000 Part-1 codec implementations.

Appendix

I. Properties of $\mathbf{P}(z)$

Proposition 4: Suppose that we have a product of the form

$$\mathbf{P}^{(N)}(z) = \begin{bmatrix} P_{0,0}^{(N)}(z) & P_{0,1}^{(N)}(z) \\ P_{1,0}^{(N)}(z) & P_{1,1}^{(N)}(z) \end{bmatrix} = \prod_{i=0}^{N-1} \mathbf{A}_i(z) \tag{25}$$

where

$$\mathbf{A}_i(z) = \begin{cases} \mathbf{J}^D \begin{bmatrix} 1 & A_i(z) \\ 0 & 1 \end{bmatrix} \mathbf{J}^D & \text{for even } i \\ \mathbf{J}^D \begin{bmatrix} 1 & 0 \\ A_i(z) & 1 \end{bmatrix} \mathbf{J}^D & \text{for odd } i, \end{cases}$$

$N \geq 0$, $D \in \{0,1\}$, $A_i(z) \not\equiv 0$ and $\mathcal{Z}^{-1}A_i(z)$ is antisymmetric about 0, for $i = 0,1,\ldots,N-1$. (Recall from Section II that $\mathbf{J}$ denotes the anti-identity matrix.) Then, $\mathbf{P}^{(N)}(z)$ must be such that

$$\mathcal{Z}^{-1}P_{0,0}^{(N)}(z) \text{ and } \mathcal{Z}^{-1}P_{1,1}^{(N)}(z) \text{ are symmetric about 0,} \tag{26a}$$

$$\mathcal{Z}^{-1}P_{0,1}^{(N)}(z) \text{ and } \mathcal{Z}^{-1}P_{1,0}^{(N)}(z) \text{ are antisymmetric about 0,} \tag{26b}$$

$$\text{for } N \neq 0\text{: } \begin{cases} \deg P_{0,0}^{(N)}(z) < \deg P_{0,1}^{(N)}(z),\ \deg P_{1,0}^{(N)}(z) < \deg P_{1,1}^{(N)}(z) & \text{for } D = 0 \\ \deg P_{0,0}^{(N)}(z) > \deg P_{0,1}^{(N)}(z),\ \deg P_{1,0}^{(N)}(z) > \deg P_{1,1}^{(N)}(z) & \text{for } D = 1, \end{cases} \tag{26c}$$

$$\text{for } N \neq 0\text{: } \begin{cases} \deg P_{0,0}^{(N)}(z) < \deg P_{1,0}^{(N)}(z),\ \deg P_{0,1}^{(N)}(z) < \deg P_{1,1}^{(N)}(z) & \text{for } (D+N) \text{ even} \\ \deg P_{0,0}^{(N)}(z) > \deg P_{1,0}^{(N)}(z),\ \deg P_{0,1}^{(N)}(z) > \deg P_{1,1}^{(N)}(z) & \text{for } (D+N) \text{ odd,} \end{cases} \tag{26d}$$

$$\deg P_{0,0}^{(N)}(z) \neq P_{0,1}^{(N)}(z), \deg P_{0,0}^{(N)}(z) \neq P_{1,0}^{(N)}(z), \deg P_{1,1}^{(N)}(z) \neq P_{0,1}^{(N)}(z), \deg P_{1,1}^{(N)}(z) \neq P_{1,0}^{(N)}(z), \tag{27}$$

$$\deg P_{0,0}^{(N)}(z) \geq 0, \deg P_{1,1}^{(N)}(z) \geq 0, \tag{28}$$

$$\text{for } i,j \in \{0,1\},\ \deg P_{i,j}^{(N)}(z) \text{ is even, except when } P_{i,j}^{(N)}(z) \equiv 0, \quad \text{and} \tag{29}$$

$$P_{0,0}^{(N)}(1) = 1,\ P_{1,1}^{(N)}(1) = 1. \tag{30}$$

Proof: The first part of the proof is by induction on N. There are two main cases to consider: $D = 0$ and $D = 1$.

In what follows, we consider the case of $D = 0$. First, consider the cases of $N = 0,1,2$. For these three cases, respectively, $\mathbf{P}^{(N)}(z)$ is given by

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ \begin{bmatrix} 1 & A_0(z) \\ 0 & 1 \end{bmatrix},\ \text{and} \begin{bmatrix} 1 & A_0(z) \\ A_1(z) & 1+A_0(z)A_1(z) \end{bmatrix}. \tag{31}$$

In the first two of the above three cases, we can easily confirm that (26) is satisfied. In the third case, (26) can also be shown to hold by using reasoning similar to that which follows. In the interest of brevity, however, we will not explicitly demonstrate this here.

To complete the inductive process, we must now show that if (26) holds for $N = K$ then (26) also holds for $N = K+1$. Thus, we begin by assuming that (26) is satisfied for $N = K$. From (25), we can write

$$\begin{aligned} \mathbf{P}^{(K+1)}(z) &\triangleq \begin{bmatrix} P_{0,0}^{(K+1)}(z) & P_{0,1}^{(K+1)}(z) \\ P_{1,0}^{(K+1)}(z) & P_{1,1}^{(K+1)}(z) \end{bmatrix} = \mathbf{A}_K(z)\mathbf{P}^{(K)}(z) \\ &= \begin{cases} \begin{bmatrix} 1 & A_K(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} P_{0,0}^{(K)}(z) & P_{0,1}^{(K)}(z) \\ P_{1,0}^{(K)}(z) & P_{1,1}^{(K)}(z) \end{bmatrix} & \text{for even } K \\ \begin{bmatrix} 1 & 0 \\ A_K(z) & 1 \end{bmatrix} \begin{bmatrix} P_{0,0}^{(K)}(z) & P_{0,1}^{(K)}(z) \\ P_{1,0}^{(K)}(z) & P_{1,1}^{(K)}(z) \end{bmatrix} & \text{for odd } K \end{cases} \\ &= \begin{cases} \begin{bmatrix} P_{0,0}^{(K)}(z)+A_K(z)P_{1,0}^{(K)}(z) & P_{0,1}^{(K)}(z)+A_K(z)P_{1,1}^{(K)}(z) \\ P_{1,0}^{(K)}(z) & P_{1,1}^{(K)}(z) \end{bmatrix} & \text{for even } K \\ \begin{bmatrix} P_{0,0}^{(K)}(z) & P_{0,1}^{(K)}(z) \\ P_{1,0}^{(K)}(z)+A_K(z)P_{0,0}^{(K)}(z) & P_{1,1}^{(K)}(z)+A_K(z)P_{0,1}^{(K)}(z) \end{bmatrix} & \text{for odd } K. \end{cases} \end{aligned}$$

Thus, the expression for $\mathbf{P}^{(K+1)}(z)$ can have one of two forms depending on the parity (i.e., evenness/oddness) of K.

Consider (26a). First, suppose that K is even. Since $P_{0,0}^{(K)}(z)$ and $A_K(z)P_{1,0}^{(K)}(z)$ both have a coefficient sequence that is symmetric about 0, their sum $P_{0,0}^{(K+1)}(z)$ must also have a coefficient sequence with the same symmetry. Thus, by observing $P_{1,1}^{(K+1)}(z) = P_{1,1}^{(K)}(z)$, we have proven that (26a) holds for $N = K+1$ when K is even. In a similar manner, we can also show that (26a) holds for $N = K+1$ when K is odd.

Consider (26b). First, suppose that K is even. Since $P_{0,1}^{(K)}(z)$ and $A_K(z)P_{1,1}^{(K)}(z)$ both have a coefficient sequence that is antisymmetric about 0, their sum $P_{0,1}^{(K+1)}(z)$ must also have a coefficient sequence with the same symmetry. Thus, by noting $P_{1,0}^{(K+1)}(z) = P_{1,0}^{(K)}(z)$, we have proven that (26b) holds for $N = K+1$ when K is even. Using similar reasoning, we can also show that (26b) holds for $N = K+1$ when K is odd.

Consider now (26c) and (26d) when K is even. Since $P_{0,0}^{(K)}(z)$ and $A_K(z)P_{1,0}^{(K)}(z)$ both have a coefficient sequence centered about 0 and $\deg P_{0,0}^{(K)}(z) < \deg P_{1,0}^{(K)}(z)$, the terms with the lowest and highest powers of z in $P_{0,0}^{(K+1)}(z)$ are contributed exclusively by $A_K(z)P_{1,0}^{(K)}(z)$. Consequently, we have

$$\deg P_{0,0}^{(K+1)}(z) = \deg A_K(z)P_{1,0}^{(K)}(z) = \deg A_K(z) + \deg P_{1,0}^{(K+1)}(z). \tag{32}$$

Since $P_{0,1}^{(K)}(z)$ and $A_K(z)P_{1,1}^{(K)}(z)$ both have a coefficient sequence centered about 0 and $\deg P_{0,1}^{(K)}(z) < \deg P_{1,1}^{(K)}(z)$, the terms with the lowest and highest powers of z in $P_{0,1}^{(K+1)}(z)$ are contributed exclusively by $A_K(z)P_{1,1}^{(K)}(z)$. Consequently, we have

$$\deg P_{0,1}^{(K+1)}(z) = \deg A_K(z)P_{1,1}^{(K)}(z) = \deg A_K(z) + \deg P_{1,1}^{(K+1)}(z). \tag{33}$$

From (32) and (33), we have $\deg P_{0,0}^{(K+1)}(z) - \deg P_{0,1}^{(K+1)}(z) = \deg P_{1,0}^{(K+1)}(z) - \deg P_{1,1}^{(K+1)}(z)$. This, however, implies that (26c) holds for $N = K+1$ when K is even. From (32), since $\deg A_K(z) > 0$, we have $\deg P_{0,0}^{(K+1)}(z) > \deg P_{1,0}^{(K+1)}(z)$, proving that the first relevant part of (26d) holds for $N = K+1$ when K is even. From (33), since $\deg A_K(z) > 0$, we have $\deg P_{0,1}^{(K+1)}(z) > \deg P_{1,1}^{(K+1)}(z)$, proving that the second relevant part of (26d) holds for $N = K+1$ when K is even.

Consider now (26c) and (26d) when K is odd. Since $P_{1,0}^{(K)}(z)$ and $A_K(z)P_{0,0}^{(K)}(z)$ both have a coefficient sequence centered about 0 and $\deg P_{0,0}^{(K)}(z) > \deg P_{1,0}^{(K)}(z)$, the terms with the lowest and highest powers of z in $P_{1,0}^{(K+1)}(z)$ are contributed exclusively by $A_K(z)P_{0,0}^{(K)}(z)$. Consequently, we have

$$\deg P_{1,0}^{(K+1)}(z) = \deg A_K(z)P_{0,0}^{(K)}(z) = \deg A_K(z) + \deg P_{0,0}^{(K+1)}(z). \tag{34}$$

Since $P_{1,1}^{(K)}(z)$ and $A_K(z)P_{0,1}^{(K)}(z)$ both have a coefficient sequence centered about 0 and $\deg P_{0,1}^{(K)}(z) > \deg P_{1,1}^{(K)}(z)$, the terms with the lowest and highest powers of z in $P_{1,1}^{(K+1)}(z)$ are contributed exclusively by

$A_K(z)P_{0,1}^{(K)}(z)$. Consequently, we have

$$\deg P_{1,1}^{(K+1)}(z) = \deg A_K(z)P_{0,1}^{(K)}(z) = \deg A_K(z) + \deg P_{0,1}^{(K+1)}(z). \tag{35}$$

From (34) and (35), we have $\deg P_{1,0}^{(K+1)}(z) - \deg P_{1,1}^{(K+1)}(z) = \deg P_{0,0}^{(K)}(z) - \deg P_{0,1}^{(K)}(z)$. This, however, implies that (26c) holds for $N = K+1$ when K is odd. From (34), since $\deg A_K(z) > 0$, we have $\deg P_{0,0}^{(K+1)}(z) < \deg P_{1,0}^{(K+1)}(z)$, proving that the first relevant part of (26d) holds for $N = K+1$ when K is odd. From (35), since $\deg A_K(z) > 0$, we have $\deg P_{0,1}^{(K+1)}(z) < \deg P_{1,1}^{(K+1)}(z)$, proving that the second relevant part of (26d) holds for $N = K+1$ when K is odd.

From above, we have shown that if (26) holds for $N = K$, then it also holds for $N = K+1$. The case of $D = 1$ can be handled in a similar fashion (but is omitted in the interest of brevity). This completes the inductive part of the proof.

Since, for $i = 0, 1, \ldots, N-1$, $\mathcal{Z}^{-1}A_i(z)$ is antisymmetric, $A_i(1) = 0$ and $\mathbf{A}_i(1) = \mathbf{I}$. Thus, $\mathbf{P}^{(N)}(1) = \mathbf{I}^N = \mathbf{I}$, and this implies that $P_{0,0}^{(N)}(1) = 1$ and $P_{1,1}^{(N)}(1) = 1$. Thus, we have that (30) holds for any N. Also, we have that (26a) and (26b) together imply (29) holds. The relationship (27) holds for $N > 0$ as a result of (26c) and (26d), while the $N = 0$ case can be trivially confirmed. Lastly, (28) holds since from earlier results we can deduce that $\deg P_{i,j}^{(N+1)}(z) \geq \deg P_{i,j}^{(N)}(z)$, and we know that $\deg P_{0,0}^{(0)}(z) \geq 0$ and $\deg P_{1,1}^{(0)}(z) \geq 0$. ■

II. PROOF OF ELASF FAMILY PROPERTIES

Proof of Property 1: First, let us consider the form of $\mathcal{Z}^{-1}H_0(z)$. For convenience, we denote the first and second terms in the expression for $H_0(z)$ in (18a) as $B(z)$ and $C(z)$, respectively (i.e., $B(z) \stackrel{\triangle}{=} \frac{1}{2}(z+1)P_{0,0}(z^2)$ and $C(z) \stackrel{\triangle}{=} (z-1)P_{0,1}(z^2)$). Due to the form of $P_{0,0}(z^2)$ (from (26a)), $B(z)$ has a coefficient sequence $b[n]$ that is symmetric about $-\frac{1}{2}$ with adjacent pairs of samples being equal in value (i.e., $b[2n] = b[2n-1]$). Likewise, due to the form of $P_{0,1}(z^2)$ (from (26b)), $C(z)$ has a coefficient sequence $c[n]$ that is symmetric about $-\frac{1}{2}$ with adjacent pairs of samples being equal in magnitude but opposite in sign (i.e., $c[2n] = -c[2n-1]$). Suppose that $P_{0,1}(z) \not\equiv 0$. In this case, from (29), (28), and (27), we know that $\deg P_{0,0}(z^2)$ and $\deg P_{0,1}(z^2)$ must differ by a nonzero integer multiple of 4. Since $H_0(z) = B(z) + C(z)$, $H_0(z)$ must have a coefficient sequence that is symmetric about $-\frac{1}{2}$ and begins and ends with pairs of coefficients that are either equal or equal in magnitude but opposite in sign. In the degenerate case, in which $P_{0,1}(z) \equiv 0$, we simply have $H_0(z) = \frac{1}{2}(z+1)$. Lastly, using an argument similar to that above, we can also show that $\mathcal{Z}^{-1}H_1(z)$ has the form stated.

Proof of Property 2: By considering the forms of $P_{0,0}(z)$ and $P_{0,1}(z)$ in (18a), we can see that

$$\deg H_0(z) = 1 + 2\max(\deg P_{0,0}(z), \deg P_{0,1}(z)). \tag{36}$$

Since, by (29), $\deg P_{0,0}(z)$ is always even and $\deg P_{0,1}(z)$ is even (except when $P_{0,1}(z) \equiv 0$), we have that $\deg H_0(z)$ is odd. Thus, H_0 is an even-length filter. Similarly, using (18b), we can show that

$$\deg H_1(z) = 1 + 2\max(\deg P_{1,1}(z), \deg P_{1,0}(z)). \tag{37}$$

Since, by (29), $\deg P_{1,1}(z)$ is always even and $\deg P_{1,0}(z)$ is even (except when $P_{1,0}(z) \equiv 0$), we have that $\deg H_1(z)$ is odd. Thus, H_1 is an even-length filter.

If at least one of the $\{\hat{A}_k(z)\}_{k=1}^{2\lambda-1}$ is not identically zero, (26c) implies that two cases are possible: 1) $\deg P_{0,0}(z) > \deg P_{0,1}(z)$ and $\deg P_{1,0}(z) > \deg P_{1,1}(z)$; or 2) $\deg P_{0,0}(z) < \deg P_{0,1}(z)$ and $\deg P_{1,0}(z) < \deg P_{1,1}(z)$. In the first case, we have from (36) and (37) that $\deg H_0(z) = 1 + 2\deg P_{0,0}(z)$ and $\deg H_1(z) = 1 + 2\deg P_{1,0}(z)$. From (27), however, we know that $\deg P_{0,0}(z) \neq \deg P_{1,0}(z)$. Therefore, $\deg H_0(z) \neq \deg H_1(z)$. In the second case, we have from (36) and (37) that $\deg H_0(z) = 1 + 2\deg P_{0,1}(z)$ and $\deg H_1(z) = 1 + 2\deg P_{1,1}(z)$. From (27), however, we know that $\deg P_{0,1}(z) \neq \deg P_{1,1}(z)$. Therefore, $\deg H_0(z) \neq \deg H_1(z)$. By combining the results for the above two cases, we have that $\deg H_0(z) \neq \deg H_1(z)$, except in the degenerate case where all of the $\{\hat{A}_k(z)\}_{k=1}^{2\lambda-1}$ are identically zero (resulting in $\deg H_0(z) = \deg H_1(z) = 1$).

Proof of Property 3: From (26b) and (30), we have

$$P_{0,1}(z^2)|_{z=\pm 1} = 0, \quad P_{1,0}(z^2)|_{z=\pm 1} = 0, \tag{38}$$

$$P_{0,0}(z^2)|_{z=\pm 1} = 1, \quad \text{and} \quad P_{1,1}(z^2)|_{z=\pm 1} = 1. \tag{39}$$

Using (38) and (39), we can deduce from (18) that $|H_0(1)| = 1$, $|H_0(-1)| = 0$, $|H_1(-1)| = 2$, and $|H_1(1)| = 0$.

Proof of Property 4: In the case of the original filter bank, the analysis polyphase matrix, $\mathbf{E}(z)$, is given by

$$\mathbf{E}(z) = \left(\prod_{k=1}^{\lambda-1} \begin{bmatrix} 1 & \hat{A}_{2k+1}(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \hat{A}_{2k}(z) & 1 \end{bmatrix}\right) \begin{bmatrix} 1 & \hat{A}_1(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}. \tag{40}$$

Suppose that we now construct a new filter bank with the lowpass and highpass analysis filters $H_0'(z)$ and $H_1'(z)$, respectively, where $H_0'(z) = \alpha_0 z F_0(z)$ and $H_1'(z) = \alpha_1 z F_1(z)$. In other words, the new analysis filters are chosen to be renormalized versions of the synthesis filters from the original filter bank. Further assume that we continue to employ the same polyphase representation for the new filter bank. Let us denote the new analysis polyphase matrix as $\mathbf{E}'(z)$. From the definition of the polyphase representation, we can show

$$\mathbf{E}'(z) = \begin{bmatrix} \alpha_0 & 0 \\ 0 & \alpha_1 \end{bmatrix} \left(\mathbf{E}^{-1}(z)\right)^T \mathbf{J}. \tag{41}$$

(Again, recall from Section II that $\mathbf{J}$ denotes the anti-identity matrix.) Substituting (40) in (41), we obtain

$$\mathbf{E}'(z) = \begin{bmatrix} \alpha_0 & 0 \\ 0 & \alpha_1 \end{bmatrix} \left(\prod_{k=1}^{\lambda-1} \begin{bmatrix} 1 & 0 \\ -\hat{A}_{2k+1}(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & -\hat{A}_{2k}(z) \\ 0 & 1 \end{bmatrix}\right) \begin{bmatrix} 1 & 0 \\ -\hat{A}_1(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \mathbf{J}. \tag{42}$$

Suppose now that we choose $\alpha_0 = \frac{1}{2}$ and $\alpha_1 = -2$. In this case, we can rewrite (42) as follows:

$$\begin{aligned}
\mathbf{E}'(z) &= \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & -2 \end{bmatrix} \left(\prod_{k=1}^{\lambda-1} \begin{bmatrix} 1 & 0 \\ -\hat{A}_{2k+1}(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & -\hat{A}_{2k}(z) \\ 0 & 1 \end{bmatrix}\right) \begin{bmatrix} 1 & 0 \\ -\hat{A}_1(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \mathbf{J} \\
&= \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & -2 \end{bmatrix} \left(\prod_{k=1}^{\lambda-1} \begin{bmatrix} 1 & 0 \\ -\hat{A}_{2k+1}(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & -\hat{A}_{2k}(z) \\ 0 & 1 \end{bmatrix}\right) \begin{bmatrix} 1 & 0 \\ -\hat{A}_1(z) & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \\
&= \left(\prod_{k=1}^{\lambda-1} \begin{bmatrix} 1 & 0 \\ 4\hat{A}_{2k+1}(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & \frac{1}{4}\hat{A}_{2k}(z) \\ 0 & 1 \end{bmatrix}\right) \begin{bmatrix} 1 & 0 \\ 4\hat{A}_1(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}.
\end{aligned}$$

Thus, the new analysis polyphase matrix, $\mathbf{E}'(z)$, (corresponding to the swapped or "transposed" filter bank) has the same general form as the original one $\mathbf{E}(z)$ (given by (40)). Therefore, the new filter bank is also associated with a transform in the ELASF family.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Hongjian Shi and the anonymous reviewers for their useful comments and suggestions which helped to improve the quality of this manuscript.

REFERENCES

[1] A. R. Calderbank, I. Daubechies, W. Sweldens, and B.-L. Yeo, "Wavelet transforms that map integers to integers," *Applied and Computational Harmonic Analysis*, vol. 5, no. 3, pp. 332–369, July 1998.

[2] H. Chao, P. Fisher, and Z. Hua, "An approach to integer wavelet transforms for lossless for image compression," in *Proc. of International Symposium on Computational Mathematics*, Guangzhou, China, Aug. 1997, pp. 19–38.

[3] M. D. Adams, *Reversible Integer-to-Integer Wavelet Transforms for Image Coding*, Ph.D. thesis, Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC, Canada, Sept. 2002, Available online from `http://www.ece.uvic.ca/~mdadams`.

[4] International Organization for Standardization and International Electrotechnical Commission, *ISO/IEC 15444-1:2000, Information technology—JPEG 2000 image coding system—Part 1: Core coding system*.

[5] International Organization for Standardization and International Electrotechnical Commission, *ISO/IEC 15444-3:2002, Information technology—JPEG 2000 image coding system—Part 3: Motion JPEG 2000*.

[6] A. Zandi, J. D. Allen, E. L. Schwartz, and M. Boliek, "CREW: Compression with reversible embedded wavelets," in *Proc. of IEEE Data Compression Conference*, Snowbird, UT, USA, Mar. 1995, pp. 212–221.

[7] A. Said and W. A. Pearlman, "An image multiresolution representation for lossless and lossy compression," *IEEE Trans. on Image Processing*, vol. 5, no. 9, pp. 1303–1310, Sept. 1996.

[8] M. D. Adams and F. Kossentini, "Reversible integer-to-integer wavelet transforms for image compression: Performance evaluation and analysis," *IEEE Trans. on Image Processing*, vol. 9, no. 6, pp. 1010–1024, June 2000.

[9] M. J. Gormish, E. L. Schwartz, A. F. Keith, M. P. Boliek, and A. Zandi, "Lossless and nearly lossless compression of high-quality images," in *Proc. of SPIE*, San Jose, CA, USA, Mar. 1997, vol. 3025, pp. 62–70.

[10] A. Cohen and J. Kovacevic, "Wavelets: The mathematical background," *Proc. of IEEE*, vol. 84, no. 4, pp. 514–522, Apr. 1996.

[11] M. J. T. Smith and S. L. Eddins, "Analysis/synthesis techniques for subband image coding," *IEEE Trans. on Acoustics, Speech, and Signal Proceesing*, vol. 38, no. 8, pp. 1446–1456, Aug. 1990.

[12] M. J. T. Smith and S. L. Eddins, "Subband coding of images with octave band tree structures," in *Proc. of IEEE ICASSP*, Dallas, TX, USA, 1987, pp. 1382–1385.

[13] M. D. Adams, F. Kossentini, and R. Ward, "Generalized S transform," *IEEE Trans. on Signal Processing*, vol. 50, no. 11, pp. 2831–2842, Nov. 2002.

[14] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," *Applied and Computational Harmonic Analysis*, vol. 3, no. 2, pp. 186–200, 1996.

[15] 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 Communications*, vol. 10, no. 1, pp. 130–137, Jan. 1992.

[16] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *Journal of Fourier Analysis and Applications*, vol. 4, pp. 247–269, 1998.

[17] M. Maslen and P. Abbott, "Automation of the lifting factorisation of wavelet transforms," *Computer Physics Communications*, vol. 127, pp. 309–326, 2000.

[18] G. Karlsson and M. Vetterli, "Extension of finite length signals for sub-band coding," *Signal Processing*, vol. 17, pp. 161–168, June 1989.

[19] K. Nishikawa, H. Kiya, and M. Sagawa, "Property of circular convolution for subband image coding," in *Proc. of IEEE ICASSP*, Mar. 1992, vol. 4, pp. 281–284.

[20] M. Iwahashi, H. Kiya, and K. Nishikawa, "Subband coding of images with circular convolution," in *Proc. of IEEE ICASSP*, Mar. 1992, pp. 1356–1359.

[21] S. A. Martucci and R. M. Mersereau, "The symmetric convolution approach to the nonexpansive implementation of FIR filter banks for images," in *Proc. of IEEE ICASSP*, Minneapolis, MN, Apr. 1993, vol. 5, pp. 65–68.

[22] H. Kiya, K. Nishikawa, and M. Iwahashi, "A development of symmetric extension methods for subband image coding," *IEEE Trans. on Image Processing*, vol. 3, no. 1, pp. 78–81, Jan. 1994.

[23] C. M. Brislawn, "Classification of nonexpansive symmetric extension transforms for multirate filter banks," *Applied and Computational Harmonic Analysis*, vol. 3, pp. 337–357, 1996.

[24] C. M. Brislawn, "Preservation of subband symmetry in multirate signal coding," *IEEE Trans. on Signal Processing*, vol. 43, no. 12, pp. 3046–3050, Dec. 1995.

[25] M. D. Adams and F. Kossentini, "Low-complexity reversible integer-to-integer wavelet transforms for image coding," in *Proc. of IEEE Pacific Rim Conference*, Victoria, BC, Canada, Aug. 1999, pp. 177–180.

[26] C. Brislawn and B. Wohlberg, "Boundary extensions and reversible implementation for half-sample symmetric filter banks," ISO/IEC JTC 1/SC 29/WG 1 N 2119, Mar. 2001.

[27] E. Majani, "Lifting implementation of even-length symmetric filters," ISO/IEC JTC 1/SC 29/WG 1 N 1914, Nov. 2000.

[28] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice-Hall, Englewood Cliffs, New Jersey, 1993.

[29] M. D. Adams and F. Kossentini, "SBTLIB: A flexible computation engine for subband transforms," ISO/IEC JTC 1/SC 29/WG 1 N 867, June 1998, Available online from `http://www.ece.uvic.ca/˜mdadams`.

[30] M. D. Adams and F. Kossentini, "JasPer: A software-based JPEG-2000 codec implementation," in *Proc. of IEEE ICIP*, Vancouver, BC, Canada, Oct. 2000, vol. 2, pp. 53–56.

[31] International Organization for Standardization and International Electrotechnical Commission, *ISO/IEC 15444-5:2002 Information technology—JPEG 2000 image coding system—Part 5: Reference software*.

[32] C. Brislawn, S. Mniszewski, M. Pal, A. Percus, B. Wohlberg, T. Acharya, P.-S. Tsai, and M. Lepley, "Even-length filter bank option (Stockholm core experiment CE03)," ISO/IEC JTC 1/SC 29/WG 1 N 2209, July 2001.