Recent results in wavelet applications

I. Daubechies

Princeton University, Princeton, New Jersey, U.S.A.

ABSTRACT

We present three recent developments in wavelets and subdivision: wavelet-type transforms that map integers to integers, with an application to lossless coding for images; rate-distortion bounds that realize the compression given by nonlinear approximation theorems for a model where wavelet compression outperforms the Karhunen-Loève approach; and smoothness results for irregularly spaced subdivision schemes, related to wavelet compression for irregularly spaced data.

Keywords: wavelets, compression, integer transforms, subdivision schemes

1. INTRODUCTION

Wavelets have emerged in the last fifteen years as a synthesis of ideas from fields as different as electrical engineering, statistics, physics, computer science and (yes!) mathematics. Wavelet transforms have beautiful and deep mathematical properties, making them a well-adapted tool for a wide range of functional spaces, or equivalently, for very different types of data. On the other hand, they can be implemented via fast algorithms, essential to convert their mathematical efficiency into a truly practical tool. There exist by now many books explaining the basics of wavelet transforms (see e.g. Refs. 1-5 to mention only a few), so we shall give only a blitz review of their properties before going on to some recent applications.

The simplest type of wavelet basis is given by dilates and translates

$$\psi_{j,k}(x) = 2^{j/2} \, \psi(2^j x - k) \ , \ j, k \in \mathbb{Z} \ , \tag{1}$$

of a single one-dimensional function $\psi(x)$, the wavelet. Only for special choices of ψ does the family $\{\psi_{j,k}\}_{j,k \in \mathbb{Z}}$ give an orthonormal basis for $L^2(\mathbb{R})$; nevertheless, many choices are possible, ranging from the most narrow solution

$$\psi(x) = \begin{array}{lll} 1 & \text{for} & 0 \leq x < 1/2 \\ -1 & \text{for} & 1/2 \leq x < 1 \\ 0 & \text{elsewhere} \end{array}$$

(the Haar wavelet) to choices for ψ that are infinitely many times differentiable, but that are much wider (namely, supported on the whole real line). Intermediate possible choices also exist, where ψ still "lives" on an interval (albeit wider than [0,1]) and is k times differentiable. All interesting wavelet bases are associated with a multiresolution analysis: there exists a partner function ϕ, the scaling function, used to build approximation spaces V_j,

$$V_j = \overline{\text{Span} \{\phi_{j,k} \ ; \ k \in \mathbb{Z}\}}, \tag{2}$$

corresponding to different resolutions (V_j has resolution 2^{-j}); the different V_j are nested in each other,

$$\cdots \subset V_j \subset V_{j+1} \subset \cdots, \tag{3}$$

and in each V_j, the $\{\phi_{j,k}\}_{k \in \mathbb{Z}}$ are an orthonormal basis; the link with the wavelet basis is given via

$$\begin{aligned} Proj_{V_{j+1}} f &= \sum_k <f, \ \phi_{j+1,k}> \phi_{j+1,k} \\ &= Proj_{V_j} f + \sum_k <f, \ \psi_{j,k}> \psi_{j,k} \, . \end{aligned} \tag{4}$$

The different nesting and orthonormality properties listed above cause the functions ϕ and ψ to be linked via relations of the type

$$\phi(x) = \sqrt{2} \sum_{n} h_n\, \phi(2x - n) \tag{5}$$

$$\psi(x) = \sqrt{2} \sum_{n} g_n\, \phi(2x - n)\,.$$

The coefficients h_n, g_n in these equations play an important role in practical implementations (in fact, they are all one really needs to implement an orthonormal wavelet transform), as well as in some mathematical constructions.

The simple scheme sketched above has been extended and generalized in many directions: biorthogonal (rather than orthonormal) bases, bases in n (rather than 1) dimensions, on an interval (rather than the whole line), on domains in $I\!R^n$ (already a more complex situation, requiring deft mathematical manipulations), using more than 1 wavelet even in 1 dimension (multiwavelets), using other dilation factors than 2, (or even dilation matrices), etc... We refer to the by now large literature or to the wavelet digest (<http://www.wavelet.org>) for more details.

In this paper we shall touch upon a few recent developments or applications. In particular, we shall talk about wavelet-like transforms that map integers to integers; about a model that illustrates why wavelet compression of stochastic processes can be superior to Karhunen-Loève approaches; and about a generalization of wavelets useful for non-uniformly spaced data.

Two of these applications make use of "lifting", a particularly simple way to implement wavelet transforms. In its simplest form, the lifting implementations correspond to a factorization of the polyphase matrix for the wavelet filters, as explained in the tutorial article by Sweldens and the author. Lifting also has, in some cases, a very interesting interpretation which makes it possible to generalize to e.g. the non-uniform setting, where the filters are time- or space-varying, and factorization is no longer applicable. Every step in the wavelet filtering splits the approximation $s_{j,l}$ of the data at multiresolution level j into a coarser approximation s_{j-1} and detail information d_{j-1} by the formulas

$$s_{j-1,l} = \sum_{k} h_{k-2n}\, s_{j,k} \quad,\quad d_{j-1,l} = \sum_{k} g_{k-2n}\, s_{j,k}\,, \tag{6}$$

which follow from (4) and (5). In the lifting interpretation, (6) is viewed as the result of several intermediate steps: the $s_{j,k}$ are split into "even" samples (to be retained, at least for the initial stage), and "odd" samples (to be replaced by information in d_{j-1}, and discarded in s_{j-1}); from the even $s_{j,2l}$ one can then compute a possible extension to all the $s_{j,k}$ (via a pre-arranged formula) which will, in places where s_j does not have a lot of structure, give quite good "guesses" for the odd $s_{j,2l+1}$; next, one looks at the difference between the "true" odd $s_{j,2l+1}$ and the "predicted" odd $s_{j,2l+1}$ – this difference is the detail information d_j ; finally, one has to adjust the even $s_{j,2l}$ to correct for aliasing, leading to the $s_{j-1,l}$. (See Ref. 7 for more details).

A few examples:

the Haar transform:

classically:

$$s_{j-1,l} = \tfrac{1}{\sqrt{2}}(s_{j,2l} + s_{j,2l+1})$$

$$d_{j-1,l} = \tfrac{1}{\sqrt{2}}(-s_{j,2l} + s_{j,2l+1})\,.$$

lifting:

$$\{s_{j,l}\} \;\rightarrow\; \begin{array}{l} e^0_{j-1,l} = s_{j,2l} \\[4pt] o^0_{j-1,l} = s_{j,2l+1} \end{array} \qquad \rightarrow\; \begin{array}{l} e^1_{j-1,l} = e^0_{j-1,l} \\[4pt] o^1_{j-1,l} = o^0_{j-1,l} - e^0_{j-1,l} \end{array}$$

$$\rightarrow\; \begin{array}{l} e^2_{j-1,l} = e^1_{j-1,l} + \tfrac{1}{2}o^1_{j-1,l} \\[4pt] o^2_{j-1,l} = o^1_{j-1,l} \end{array} \qquad \rightarrow\; \begin{array}{l} s_{j-1,l} = \sqrt{2}\,e^2_{j-1,l} \\[4pt] d_{j-1,l} = \tfrac{1}{\sqrt{2}}o^2_{j-1,l}\,. \end{array}$$

the 2-2 biorthogonal transform:

classically:

$$s_{j-1,l} = \tfrac{1}{\sqrt{2}}(-\tfrac{1}{4}s_{j,2l-2} + \tfrac{1}{2}s_{j,2l-1} + \tfrac{3}{2}s_{j,2l} + \tfrac{1}{2}s_{j,2l+1} - \tfrac{1}{4}s_{j,2l+2})$$

$$d_{j-1,l} = \tfrac{1}{\sqrt{2}}(-\tfrac{1}{2}s_{j,2l} + s_{j,2l+1} - \tfrac{1}{2}s_{j,2l+2}).$$

lifting:

$$s_{j,l} \quad \rightarrow \quad e^0_{j-1,l} = s_{j,2l} \qquad\qquad \rightarrow \quad e^1_{j-1,l} = e^0_{j-1,l}$$

$$o^0_{j-1,l} = s_{j,2l+1} \qquad\qquad\qquad o^1_{j-1,l} = o^0_{j-1,l} - \tfrac{1}{2}(e^0_{j-1,l} + e^0_{j-1,l+1})$$

$$\rightarrow \quad e^2_{j-1,l} = e^1_{j-1,l} + \tfrac{1}{4}(o^1_{j-1,l-1} + o^1_{j-1,l}) \quad \rightarrow \quad s_{j-1,l} = \sqrt{2}\, e^2_{j-1,l}$$

$$o^2_{j-1,l} = o^1_{j-1,l} \qquad\qquad\qquad\qquad d_{j-1,l} = \tfrac{1}{\sqrt{2}} o^2_{j-1,l}.$$

In both cases, we have one "predict" and one "update" step; for other wavelet transforms, more such alternating steps may be needed. Note that the "triangular" nature of the lifting operations, (in which at every stage only the "evens" or the "odds" are modified, using information only from the other stream) makes it trivial to invert the transform. All these features will be useful in our first application.

2. WAVELET-LIKE TRANSFORMS THAT MAP INTEGERS TO INTEGERS

This section is based on joint work with A.R. Calderbank, W. Sweldens and B.L. Yeo.[8] Integer transforms are especially interesting for lossless coding. With standard wavelet filters, the filtered outputs do not consist of integers, even if the input data consist of sequences of integers, as is the case for images. Yet, for lossless coding of images it would be of interest to be able to characterize the output completely again by integers.

Let us look first at the Haar transform. Even if we "renormalize" by $\sqrt{2}$, defining

$$\begin{aligned} s_{j-1,l} &= \tfrac{1}{2}(s_{j,2l} + s_{j,2l+1}) \\ d_{j-1,l} &= s_{j,2l+1} - s_{j,2l}, \end{aligned} \qquad (7)$$

this is not an integer transform, because of the division by two. Obviously we can build an integer version by simply omitting the division by two in the formula for $s_{1,l}$ and calculating the sum instead of the average. But a more efficient construction, known as the S transform (Sequential), is possible.[9,10] Several definitions of the S transform exist, differing only in implementation details. An easy way to construct the S transform is to start from the lifting form of (7):

$$\begin{aligned} d_{j-1,l} &= s_{j,2l+1} - s_{j,2l} \\ s_{j-1,l} &= s_{j,2l} + \tfrac{1}{2}d_{j-1,l}. \end{aligned} \qquad (8)$$

Now we can change (8) into an integer transform by truncating the division in the second step:

$$\begin{aligned} d_{1,l} &= s_{0,2l+1} - s_{0,2l} \\ s_{1,l} &= s_{0,2l} + \lfloor d_{1,l}/2 \rfloor. \end{aligned}$$

Even though the transform now is non-linear, lifting allows us to find immediately the inverse: reverse the order and change the signs of the forward transform:

$$\begin{aligned} s_{0,2l} &= s_{1,l} - \lfloor d_{1,l}/2 \rfloor \\ s_{0,2l+1} &= d_{1,l} + s_{0,2l}. \end{aligned}$$

So we can obtain an integer transform using simply truncation and without losing invertibility. Two generalizations of the S transform have been proposed: the TS transform[11] and the S+P transform.[12] The idea behind both is the same: add a third stage in which a prediction of the difference is computed based on the average values; the new high pass coefficient is now the difference of this prediction and the actual difference. This can be thought of as another lifting step and therefore immediately results in invertible integer wavelet transforms.

4

The TS transform turns out to be an integer version of the (3, 1) biorthogonal wavelet transform of Cohen-Daubechies-Feauveau.[13] In that transform, one has

$$
\begin{aligned}
d_{1,l}^{(1)} &= s_{0,2l+1} - s_{0,2l} \\
s_{1,l} &= s_{0,2l} + d_{1,l}^{(1)}/2 \\
d_{1,l} &= d_{1,l}^{(1)} + s_{1,l-1}/4 - s_{1,l+1}/4 \, .
\end{aligned}
\tag{9}
$$

The coefficients of the prediction are chosen so that in case the original sequence was a second degree polynomial in l, then the new wavelet coefficient $d_{1,l}$ is exactly zero. The integer version or TS transform can now be obtained by truncating the non-integers coming from the divisions in the last two steps:

$$
\begin{aligned}
d_{1,l}^{(1)} &= s_{0,2l+1} - s_{0,2l} \\
s_{1,l} &= s_{0,2l} + \lfloor d_{1,l}^{(1)}/2 \rfloor \\
d_{1,l} &= d_{1,l}^{(1)} + \lfloor s_{1,l-1}/4 - s_{1,l+1}/4 + 1/2 \rfloor \, .
\end{aligned}
\tag{10}
$$

In the last step we add a term $1/2$ before truncating to avoid bias; in the second step this is not needed because the division is only by two. As the transform is already decomposed into lifting steps, the inverse can again be found immediately by reversing the equations and flipping the signs:

$$
\begin{aligned}
d_{1,l}^{(1)} &= d_{1,l} - \lfloor s_{1,l-1}/4 - s_{1,l+1}/4 + 1/2 \rfloor \\
s_{0,2l} &= s_{1,l} = \lfloor d_{1,l}^{(1)}/2 \rfloor \\
s_{0,2l+1} &= d_{1,l}^{(1)} + s_{0,2l} \, .
\end{aligned}
\tag{11}
$$

In the S+P transform (**S** transform + **P**rediction), the predictor for $d_{1,l}^{(1)}$ does not only involve $s_{1,k}$ values but also a previously calculated $d_{1,l+1}^{(1)}$ value. The general form of the transform is

$$
\begin{aligned}
d_{1,l}^{(1)} &= s_{0,2l+1} - s_{0,2l} \\
s_{1,l} &= s_{0,2l} + \lfloor d_{1,l}^{(1)}/2 \rfloor \\
d_{1,l} &= d_{1,l}^{(1)} + \lfloor \alpha_{-1}(s_{1,l-2} - s_{1,l-1}) + \alpha_0(s_{1,l-1} - s_{1,l}) + \alpha_1(s_{1,l} - s_{1,l+1}) - \beta_1 \, d_{1,l+1}^{(1)} \rfloor \, .
\end{aligned}
\tag{12}
$$

Note that the TS transform is a special case, namely when $\alpha_{-1} = \beta_1 = 0$ and $\alpha_0 = \alpha_1 = 1/4$. Said and Pearlman examine several choices for (α_w, β_1) and in the case of natural images suggest $\alpha_{-1} = 0$, $\alpha_0 = 2/8$, $\alpha_1 = 3/8$ and $\beta_1 = -2/8$. It is interesting to note that, even though this was not their motivation, this choice without truncation yields a high pass analyzing filter with 2 vanishing moments.

Therefore, the lossless S, TS and S+P transforms can all be viewed as special cases of the lifting schemes, truncating appropriately.

One can obviously also do this for the decompositions into lifting steps of other wavelet transforms. Two particularly interesting examples are the following two transforms:

(4,2) interpolating transform

$$
d_{1,l} = s_{0,2l+1} - \lfloor 9/16(s_{0,2l} + s_{0,2l+2}) - 1/16(s_{0,2l-2} + s_{0,2l+4}) + 1/2 \rfloor
$$

$$
s_{1,l} = s_{0,2l} + \lfloor 1/4(d_{1,l-1} + d_{1,l}) + 1/2 \rfloor \, .
$$

(2+2,2) transform

$$
d_{1,l}^{(1)} = s_{0,2l+1} - \lfloor 1/2(s_{0,2l} + s_{0,2l+2}) + 1/2 \rfloor
$$

$$
s_{1,l} = s_{0,2l} + \lfloor 1/4(d_{1,l-1}^{(1)} + d_{1,l}^{(1)}) + 1/2 \rfloor
$$

$$
d_{1,l} = d_{1,l}^{(1)} - \lfloor \alpha(-1/2 \, s_{1,l-1} + s_{1,l} - 1/2 \, s_{1,l+1})
$$

$$
+ \beta(-1/2 \, s_{1,l} + s_{1,l+1} - 1/2 \, s_{1,l+2}) + \gamma d_{1,l+1}^{(1)} + 1/2 \rfloor \, .
$$

Without truncation, we want the scheme to have 4 vanishing moments. This leads to the following conditions:

$$8\beta + 3\gamma = 1$$

$$4\alpha + 4\beta + \gamma = 1 \,.$$

Special cases are: (1) $\alpha = 1/6, \beta = 0, \gamma = 1/3$, (2) $\alpha = 1/8, \beta = 1/8, \gamma = 0$ and, (3) $\alpha = 1/4, \beta = -1/4, \gamma = 1$. In our experiments we found that (2) works considerably better than (1) and (3), and this is what we use in practice.

These wavelet-like transforms, combined with an arithmetic coder, were used for lossless coding in Ref. 8, with results that often improved on S+P, but that were still short of the best lossless compression results, obtained via non-multiresolution methods. More recently these results were pushed further: work by N. Memon, X. Wu and B.L. Yeo[14] shows how the use of context information in the coding improves these results, so that they become comparable to the state of the art.

3. WAVELET COMPRESSION $\neq$ KARHUNEN-LOÈVE APPROXIMATION

This section is based upon joint work with A. Cohen, O. Guleryuz and M. Orchard.[15] In theoretical models for the mathematical study of compression, signals and particularly images are often viewed as realizations of an (unknown) stochastic process. The corresponding Karhunen-Loève basis (KL), as the orthonormal basis that optimally decorrelates this process, i.e. the basis $(\varphi_n)_{n \in \mathbb{N}}$ that minimizes $E(\|s - \sum_{h=1}^{N} \langle s, \varphi_n \rangle \varphi_n\|^2)$ for every N, is then viewed as the best possible basis to compress the signals or images. In practice, determining this KL basis exactly may be cumbersome and computationally intensive, suggesting the use of a basis that is easier to work with and that is still "close" to the KL basis, in the sense that it also decorrelates well (although not optimally). This has been argued as a justification both for DCT methods and for wavelet transforms.

Although the usefulness of KL-bases is well documented and beyond dispute in many applications, there has been a growing realization that optimizing decorrelation for the stochastic process may not be the final or even the most important point in signal compression. In the terms of mathematical approximation theory, this corresponds to a shift from linear approximation to non-linear approximation.

In linear approximation theory, given an orthonormal basis of functions $\varphi_1, \varphi_2, \ldots$, one seeks to estimate, as a function of N, the truncation error

$$e_N(f) = \|f - \sum_{n=1}^{N} \langle f, \varphi_n \rangle \varphi_n\| \,. \tag{13}$$

The behavior of $e_N(f)$ as N increases gives us information about f and vice versa. For instance, if the φ_n are either the Fourier basis on $[0, 1]$ or a wavelet basis (with its logical ordering), and if $\| \ \|$ in (13) is taken to be the L^2-norm, $\|g\| = \left(\int_0^1 |g(t)|^2 dt\right)^{1/2}$, then the decay of $e_N(f)$ characterizes the smoothness of f in an L^2-sense: for instance, $e_N(f) \leq CN^{-k-\varepsilon}$ implies that f and its first k derivatives are all in L^2. The error $e_N(f)$ can be rewritten as

$$e_N(f) = \text{dist}_{L^2}(f, S_N) \,,$$

where S_N is the linear vector space spanned by the first N basis functions,

$$S_N = \text{Span}(\varphi_1, \ldots, \varphi_N) = \left\{ \sum_{n=1}^{N} c_n \varphi_n; c_n \in \mathbb{C} \right\} \,.$$

The KL basis for a stochastic process fits within this linear approximation theory framework: it is the basis for which $E\left(\text{dist}_{L^2}(s, S_N)^2\right)$ is minimized, for every N.

Non-linear approximation of a function f, with the same orthonormal basis $\varphi_1, \varphi_2, \ldots$ as before, seeks to estimate, as N increases, the decay of the distance between f and the best possible approximation to f by a linear combination of φ_n that uses at most N terms (as opposed to the best linear combination with the **first** N basis functions, as before). That is, we now have

$$e_N = \text{dist}(f, \mathbf{S}_N)$$

$$\mathbf{S}_n = \left\{ \sum_n c_n \varphi_n; c_n \in \mathbb{C}, \#\{n; c_n \neq 0\} \leq N \right\} \,.$$

The set $\mathbf{S}_N$ is no longer a linear space (since the sum of two arbitrary elements of $\mathbf{S}_N$ generally uses more that N basis functions), hence the name *non-linear* approximation. Non-linear approximation has been studied in detail by DeVore, Jawerth, and Popov; see e.g. their paper Ref. 16 for a review. Typically, different approximation estimates are obtained by non-linear approximation than by linear approximation with respect to the same basis. Or, turning this upside down, different behavior of f can be characterized by the same decay behavior as $N \to \infty$ of the truncation error in non-linear approximation versus linear approximation. Another manifestation of this difference is that for stochastic processes, the KL basis need not be the basis that minimizes the non-linear approximation error.

The difference between linear and non-linear approximation was illustrated in Ref. 17 where a 1-dimensional model of piecewise smooth processes, inspired by images, was analyzed. For this toy model, it was shown that the expected non-linear approximation error using a wavelet basis is asymptotically superior (with a decay at least as fast as CN^{-1} for $N \to \infty$) to the best expected linear approximation error (this error, with respect to the KL basis, is bounded below by $CN^{-1/2}$ for $N \to \infty$), even though wavelets are not the KL basis for the stochastic process.

The success of wavelet bases in non-linear approximation, as analyzed in Ref. 18 and illustrated by the piecewise smooth toy model in Ref. 17 was interpreted by mathematicians to be the "true" reason for the usefulness of wavelets in signal compression, rather than their potential for decorrelation. Yet non-linear approximation estimates are still a long way from a mathematical analysis that would be directly related to compression issues; the practice is **not**, as suggested by non-linear approximation, to "compress" all the information into N coefficients, discarding all the other information, and checking how well one does. A model closer to practice would estimate the error or distortion given that all the information (truncated coefficients as well as the choice of which indices to retain) has to fit within a certain bit budget. Such rate-distortion bounds are closer to the practice of coding for compression, and therefore more convincing from the point of view of the electrical engineer.

On the other hand, coding techniques for compression have become increasingly sophisticated in the last few years, steadily obtaining better results. Some of the improved strategies were inspired by, or can be heuristically explained by mathematical arguments. Yet there exist no mathematical estimates for these coding strategies of the same level of detail and depth as for the coding-wise more naïve non-linear approximation theory.

In Ref. 15 we prove that the non-linear approximation results of Ref. 17 translate into the corresponding rate-distortion bounds. In the case of nonlinear approximation, where the wavelet coefficients are "re-ordered" according to how significant they are, we need to evaluate the bit rate necessary to encode the addressing of the different types of coefficients. The bit rate comes thus in two parts: bit rate for the coefficients, and bit rate for the addressing. Unless good coding strategies are used, it turns out that the addressing bit rate is dominant, and spoils the compression given by nonlinear approximation theorems. It turns out that smart coding techniques <u>can</u> achieve the optimal rate predicted by nonlinear approximation theory. In particular, we show that two coding strategies, which are applied in practice (namely, a model which codes the coefficients as coming from a mixture of two distributions; and a tree-coder) do achieve a rate-distortion band of the type

$$D \leq C\,R^{-\alpha},$$

where α is the smoothness of the pieces in the piecewise smooth model of Ref. 17.

4. WAVELETS FOR NON-UNIFORMLY SAMPLED DATA

This section concerns joint work with I. Guskov and W. Sweldens.[19] When wavelet bases are constructed via a lifting scheme, as described above, the computation of the wavelet coefficients consists of a prediction step for the "odds" from the "evens", and a comparison of the true "odds" with these predictions. If the wavelet coefficients are zero – i.e. if we are considering a scaling function – then the predictions are exact at all levels: to build a scaling function for this scheme one thus needs only to iterate the prediction scheme level after level, generating an increasingly finer sampling of the scaling function through a subdivision scheme. This approach (used, in fact, to plot all compactly supported wavelets and scaling functions in e.g. Ref. 1) is not limited to the case where the sampling points are uniformly distributed. Two types of non-uniform cases can be considered. In the semi-regular case, the original samples ("at level 0") are not equally spaced, but the subdivision scheme still introduces new grid points midway between old ones. This scheme is used in computer graphics applications, where subdivision is applied to generate smooth curves or surfaces. In the irregular case, new grid points need not be in the middle between old points, even at infinity. This irregular setting comes up naturally in the case of compression of, or multiresolution analysis for,

irregular samples. The user provides data, sampled on a closely spaced but irregular grid, which one can think of as the "finest" level grid. Resampling onto a regular grid is typically costly and may generate unwanted artifacts. One can then build a multiresolution analysis and an associated wavelet transform for the irregular grid, using the lifting scheme, leading to spatially variant filters.[7]

It is then important to understand how smooth the wavelets are for these irregular scheme. As in the uniformly sampled scheme, this smoothness is governed by that of the scaling functions, so that the question reduces to the study of smoothness for irregularly spaced subdivision. Outside the context of spline functions, where there exists a large body of work on splines for irregularly spaced knots, work on subdivision schemes for non-equally spaced grids started only recently. The semi-regular case is completely understood in 1D,[20] and also reasonably well understood in the much harder higher dimensional setting.[21-23] The completely irregular case is much trickier. In Ref. 19 we develop techniques suitable for this more general case, and we concentrate on determining the smoothness for the so-called four-point scheme. In the regular setting, this interpolation scheme was first introduced in Ref. 24 and Ref. 25. It consists in taking for the function value at a "new point" the value of the cubic that is completely determined by the two nearest neighboring "old points" both on the left and on the right and their function values. In the regular setting, where one starts from data at the integers (i.e. in $\mathbb{Z}$), then computes interpolated values at $\mathbb{Z}/2\backslash\mathbb{Z}$ in a first stage, followed by an interpolation to find the values at $\mathbb{Z}/4\backslash\mathbb{Z}/2$ in a second stage, etc..., the cubic interpolation rule leads to

$$f(x_{j,2k+1}) = \frac{9}{16}[f(x_{j-1,k}) + f(x_{j-1,k+1})] - \frac{1}{16}[f(x_{j-1,b-1}) + f(x_{j-1,k+2})], \tag{14}$$

where $x_{j,l} = l2^{-j}$. In the completely irregular setting, the $x_{j,k}$ are not given by this simple formula (although we still view the process as a "refinement", so that $x_{j,2k} = x_{j-1,k}$), and (14) is replaced by a Lagrange interpolation formula with coefficients that depend on the $x_{j-1,k}$. In order to avoid refinement of our grid that would be too pathological, we impose some conditions; introducing $d_{j,k} = x_{j,k+1} - x_{j,k}$, we say the refinement procedure is <u>homogeneous</u> if, for some $\gamma < \infty$ independent of j, k,

$$\max(d_{j,k+1}, d_{j,k+1}) \leq \gamma d_{j,k} \quad \text{for all } j, k;$$

we say the procedure is <u>dyadically balanced</u> if, for some $\alpha < \infty$ independent of j, k,

$$\max(d_{j,2k}, d_{j,2k+1}) \leq \alpha \min(d_{j,2k}, d_{j,2k+1}) \quad \text{for all } j, k.$$

The latter condition says that whenever an interval $[x_{j-1,k}, x_{j-1,k+1}]$ is split by introducing a new point $x_{j,2k+1}$, the split cannot be too lopsided; $\alpha = 1$ reduces to the regular case. A homogeneous refinement procedure is always dyadically balanced, but the reverse is not true. (See Ref. 19 for examples and a discussion.)

We prove in Ref. 19 that the refinement scheme based upon cubic interpolation always leads to a limit function that is at least C^1 if the multi-level grid is dyadically balanced. For $\alpha < 2$, we even find that the limit function f is almost C^2, in the sense that

$$|f'(x + t) - f'(x)| \leq C|t| \, |log|t||,$$

a result that is well-known for the regular case, where it is optimal. The proof is fairly technical, and quite intricate for the general dyadically balanced case. For the slightly less general homogeneous case, it simplifies considerably. (See Ref. 19 for all details.)

ACKNOWLEDGMENTS

REFERENCES

1. I. Daubechies, "Ten lectures on wavelets," in *CBMS conference on wavelets, Society for Industrial and Applied Mathematics* **61**, 1992.
2. G. Strang and T. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge Press, Wellesley, MA, 1996.
3. S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 1998.
4. M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*, Prentice Hall, Englewood Cliffs, NJ, 1995.

5. C. K. Chui, "An introduction to wavelets," in *Wavelet Analysis and its Applications*, vol. I, Academic Press, 1992.

6. I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *submitted to Journal of Fourier Analysis*.

7. W. Sweldens, "The lifting scheme: A construction of second generation wavelets," *to appear in SIAM J. Math. Anal.*

8. A. R. Calderbank, I. Daubechies, W. Sweldens, and B.-L. Yeo, "Wavelet transforms that map integers to integers," *submitted to Applied and Computational Harmonic Analysis*.

9. M. Rabbini and P. W. Jones, *Digital Image Compression Techniques*, SPIE Opt. Eng. Press, Bellingham, WA, 1991.

10. V. K. Heer and H.-E. Reinfelder, "A comparison of reversible methods for data compression," in *Medical Imaging IV*, *Proc. SPIE* **1233**, pp. 354–365, 1990.

11. A. Zandi, M. Boliek, E. L. Schwartz, and M. J. Gormish, "Crew lossless/lossy medical image compression," *RICOH California Research Center* **Technical Report CRC-TR-9526**, 1995.

12. A. Said and W. A. Pearlman, "An image multiresolution representation for lossless and lossy image compression," *IEEE Trans. Image Process.* **5(9)**, pp. 1303–1310, 1996.

13. A. Cohen, I. Daubechies, and J. Feauveau, "Bi-orthogonal bases of compactly supported wavelets," *Comm. Pure Appl. Math.* **45**, pp. 485–560, 1992.

14. N. Memon, X. Wu, and B.-L. Yeo, "Entropy coding techniques for lossless image compression with reversible integer wavelet transforms," in *International Conference on Image Processing, IEEE Signal Processing Society*, 1997.

15. A. Cohen, I. Daubechies, and O. Guleryuz, "On the importance of combining wavelet-based non-linear approximation with coding strategies," *Princeton University, preprint*, 1997.

16. R. A. DeVore, B. Jawerth, and V. Popov, "Compression of wavelet decompositions," *Americ. J. of Math.* **114**, pp. 737–785, 1992.

17. A. Cohen and J. P. D'Ales, "Non-linear approximation of random functions," *Siam J. of Appl. Math* **57-2**, pp. 518–540, 1997.

18. R. A. DeVore, B. Jawerth, and B. J. Lucier, "Image compression through wavelet transform coding," *IEEE Trans. Info. Theory* **38(2)**, pp. 719–746, 1992.

19. I. Daubechies, I. Guskov, and W. Sweldens, "Regularity of irregular subdivision," *to appear in Constr. Approx*.

20. J. Warren, "Binary subdivision schemes for functions over irregular knot sequences," in *Mathematical Methods in CAGD III*, M. Daehlen, T. Lyche, and L. Schumaker, eds., Academic Press, 1995.

21. U. Reif, "A unified approach to subdivision algorithms near extraordinary vertices," *Comput. Aided Geom. Des.* **12**, pp. 153–174, 1995.

22. J. Warren, "Subdivision methods for geometric design," *Unpublished manuscript* <http://www.cs.rice.edu/~jwarren> .

23. D. Zorin, "C^k continuity of subdivision surfaces," *California Institute of Technology* **Technical Report**, 1996.

24. S. Dubuc, "Interpolation through an iterative scheme," *J. Math. Anal. Appl.* **114**, pp. 185–204, 1986.

25. N. Dyn, D. Levin, and J. Gregory, "A 4-point interpolatory subdivision scheme for curve design," *Comput. Aided Geom. Des.* **4**, pp. 257–268, 1987.