UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

Multiresolution Based on Weighted Averages of the Hat Function

F. Arandiga
Rosa Donat
Ami Harten

September 1993

CAM Report 93-34

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555

MULTIRESOLUTION BASED ON WEIGHTED AVERAGES OF THE HAT FUNCTION

F. ARÀNDIGA[1]*, ROSA DONAT[1]† AND AMI HARTEN[2]‡

Abstract. We study the properties of the multi-resolution analysis corresponding to discretization by local averages with respect to the hat function. We describe the reconstruction procedures consistent with this multi-resolution setting and the associated prediction operators that allow us to climb up the ladder from coarse to finer levels of resolution. Data-dependent (i.e. linear) as well as data-independent (i.e. non linear) reconstruction operators are described. Linear reconstruction techniques allow us, under certain circumstances, to construct a basis of generalized wavelets for the multi-resolution representation of the original data. Non linear reconstruction techniques lead to very efficient compression algorithms for functions with singularities, but need an error-control mechanism to ensure the stability of the compression algorithm.

* Research supported by ONR-N00014-92-J-1890

† Research supported by DGICYT PS90-0265 and ONR-N00014-92-J-1890

‡ Research supported by ONR-N00014-92-J-1890 and NSF-DMS91-03104

[1] Departament de Matemàtica Aplicada. Universitat de València (Spain)

[2] School of Mathematical Sciencies. Tel-Aviv University and Departament of Mathematics UCLA.

1. Introduction. Consider the set of weighted averages of a periodic function $f(x)$ of period 1 corresponding to a uniform partition of $[0,1]$ of size h_k, i.e.

$$\bar{f}_j^k = < f, \frac{1}{h_k}\omega(\frac{x - x_j^k}{h_k}) >, \quad j = 1, \ldots, J_k \qquad J_k = 1/h_k = 2^k J_0$$

where the scaling function $\omega(x)$ is a function of compact support that satisfies

$$\int \omega(x)dx = 1.$$

If the scaling function satisfies a dilation relation of the type

$$\omega(y) = 2\sum_l \alpha_l \omega(2y - l),$$

then the sequence $\left\{\{\bar{f}_j^k\}_{j=1}^{J_k}\right\}_{k=0}^{+\infty}$ of weighted averages on the sequence of nested dyadic grids $X^k = \{x_j^k\}, \quad x_j^k = j.h_k, \quad j = 1, \ldots, J_k$ constitutes a *discrete multi-resolution analysis* (see [11, 13]) in the sense that knowledge of the discrete values at a certain level of resolution determines the corresponding values at all coarser levels, in fact the dilation relation for the scaling function implies that

$$\bar{f}_j^k = \sum_l \alpha_l \bar{f}_{2j+l}^{k+1},$$

that is, $\bar{f}^k$, and therefore the local averages at all larger scales, can be evaluated from $\bar{f}^{k+1}$ without explicit knowledge of the function itself.

The multi-resolution analysis can also be viewed as a multi-scale decomposition of the original signal. The key role here is played by the reconstruction procedure. To a given discretization $\mathcal{D}_k f = \{\bar{f}_j^k\}$ one associates a reconstruction operator $\mathcal{R}_k$ that approximates the given function from its discrete values on the k-th level of resolution. For consistency, the reconstruction operators must satisfy

$$\mathcal{D}_k \mathcal{R}_k = I \qquad \text{i.e.} \qquad < \mathcal{R}_k(x, \bar{f}^k), \frac{1}{h_k}\omega(\frac{x - x_j^k}{h_k}) >= \bar{f}_j^k$$

Defining $Q_k(x; f) = \mathcal{R}_k(x; \bar{f}^k) - \mathcal{R}_{k-1}(x; \bar{f}^{k-1})$, we can write

$$\mathcal{R}_L(x; \bar{f}^L) = \mathcal{R}_0(x; \bar{f}^0) + \sum_k Q_k(x, f).$$

Then, the multi-resolution process can also be viewed as the computation of succesively coarser approximations to the original signal, together with the difference in information between every two succesive levels.

At each level of resolution we can use the reconstruction procedure to predict $f(x)$ and its weighted averages at the next finer level of resolution

$$\tilde{f}^{k+1} = \mathcal{D}_{k+1}\mathcal{R}_k \bar{f}^k.$$

The prediction errors

$$e^k = \tilde{f}^k - \bar{f}^k$$

measure our ability to predict $\mathcal{D}_k f = \bar{f}^k$ from our knowledge of $\mathcal{D}_{k-1} f = \bar{f}^{k-1}$. Large prediction errors at the $k-th$ level indicate that new scales, not predictable from the knowledge of the low resolution levels by our approximation method, are present at the $k-th$ level.

It is easy to see that

$$\mathcal{D}_{k-1}\mathcal{R}_k e^k = 0 \qquad \text{i.e.,} \qquad \sum_l \alpha_l e^k_{2j+l} = 0$$

which means that only half of these values are independent. Removing this redundancy, we obtain the *scale coefficients*, $d^k(f)$. There is a one-to-one correspondence between $\mathcal{D}_k f$ and its *multi-resolution representation*, i.e. the set $(\bar{f}^0, d^1(f), \cdots, d^k(f))$. The scale coefficients are directly related to the prediction errors, if these are small at a certain location on a given scale it means that f is properly resolved on the particular scale at that location, thus they can be set to zero, reducing the dimensionality of the discrete representation with no significant alteration of the information contents.

This observation is the basis of the data compression algorithms associated to multi-resolution settings.

Then, if a numerical problem can be embeded in a multi-resolution setting, we can improve the efficiency of the numerical solution algorithm by applying data compression to the numerical solution ([3, 14, 15]) as well as to the multi-resolution representation of the solution operator ([5, 9, 2]). We can also reorganize the numerical solution algorithm as a multiscale computation, where we solve the problem directly only in the coarsest level of resolution and then advance from coarse to fine levels by prediction and correction ([5, 12]).

The choice of the scaling function is often dictated by the nature of the computational problem. Recall that the discretization and reconstruction operators act on a space of functions $\mathcal{F}$:

$$\begin{aligned} \mathcal{D}_k &: \mathcal{F} \longrightarrow V_k \\ \mathcal{R}_k &: V_k \longrightarrow \mathcal{F} \end{aligned}$$

Usually, the property $\mathcal{D}_k\mathcal{R}_k = I$ determines the *natural* function space for the multi-resolution setting.

The simplest discretization procedure is by pointvalues, this corresponds to $\omega = \delta$, Dirac's δ-function. The reconstruction procedure in this case is given by any interpolation method (see [11, 13]) of the pointvalued data. The *natural* function space $\mathcal{F}$ is, thus, that of continuous functions.

When the weight function is the box function $\omega = \chi_{[0,1]}$, the discretization operators compute the cell averages of the given function on each one of the grids X^k. The reconstruction procedures have to be *conservative*, i.e., their cell averages at each level of resolution coincide with those of f. Since the function always appears under the integral sign, the natural function space is that of piecewise smooth functions with jump discontinuities.

The interpolatory and cell-average multi-resolution settings have been extensively studied in [11, 13]. We refer to those two papers for a complete description and to [13, 14, 15, 2, 1, 7] for applications to various numerical problems. The point-value multi-resolution is most appropriate when dealing with numerical problems that have continuous solutions. The cell average formulation has been used succesfully in designing fast algorithms for the numerical

approximation to solutions of conservation laws, which are piecewise smooth functions with at most jump discontinuities.

The purpose of this paper is to study the multi-resolution setting corresponding to choosing the hat function, $\omega = \chi_{[-1,0]} * \chi_{[0,1]}$ as the scaling function. This multi-resolution setting is most suitable for $\mathcal{F}$ of piecewise-smooth distributions, i.e. piecewise smooth functions with a finite number of δ-singularities. The space of piecewise smooth distributions is used in vortex methods for the numerical solution of fluid dynamics problems. We expect that the hat-averaged multi-resolution set-up will be useful for this type of problems.

The paper is organized as follows: Sections 2 and 3 review the basic facts about the multi-resolution analysis and multi-resolution representation of a set of discrete data. A more extensive description can be found in [11] and [13].

In section 4 we highlight the main points of the interpolatory and cell average multi-resolution, which are described in more detail in [11] and [13], for comparison and reference purposes.

The core of the paper starts in section 5, where we describe the hat-weighted multi-resolution and characterize the reconstruction procedures that are consistent with this multi-resolution analysis. Reconstruction operators that are data-independent are also linear functionals. They add a rich functional structure to the multi-resolution setting and the multi-resolution representation, under certain circumstances, admits a pseudo-wavelets basis of functions. This ground is covered in section 6, along with a detailed analysis of two data-independent reconstruction procedures.

Section 7 is devoted to data compression techniques and their stability properties. Data-dependent reconstruction operators are non linear functionals. They lead to efficient compression techniques, however one must be careful to ensure the stability of the compression algorithm. In this section we describe a modified encoding-decoding algorithm that leads to a stable data compression algorithm.

In section 8, we report several numerical experiments that confirm our theoretical results. We also compare the efficiency and accuracy of data compression algorithms based on multi-resolution analysis of point-values, cell-averages and hat-averages.

2. Multiresolution analysis. We consider the interval $[0,1]$ partitioned into $J_L = 2^L \cdot J_0$ intervals of size $h_L = 1/J_L$ by $x_j^L = j.h_L$, $j = 1, \ldots, J_L$. To simplify our presentation let us consider a periodic function $f(x)$ with period 1, and assume that f is discretized on this grid by local averages,

$$\bar{f}_j^L = < f, \frac{1}{h_L}\omega(\frac{x - x_j^L}{h_L}) >, \quad j = 1, \ldots, J_L \tag{1}$$

where $<,>$ is the Euclidean inner product and $\omega(x)$, the weight function, is a function of compact support satisfying

$$\int \omega(x)dx = 1. \tag{2}$$

We construct the set of nested dyadic grids $\{X^k\}$, $k \geq 0$ of size $h_k = 2^{-k}h_0$ with $J_k = 2^k \cdot J_0$ intervals by

$$X^k = \{x_j^k\} \qquad x_j^k = j.h_k, \quad j = 1, \ldots, J_k \tag{3}$$

Thus, $x_{2j}^k = x_j^{k-1}$.

With each of the grids we associate a discretization $\{\bar{f}_j^k\}_{j=1}^{J_k}$ of the function $f(x)$

$$(\mathcal{D}_k f)_j = \bar{f}_j^k =< f, \frac{1}{h_k}\omega(\frac{x - x_j^k}{h_k}) >\equiv< f, \omega_j^k >, \quad j = 1, \ldots, J_k \tag{4}$$

where ω_j^k are scaled translates of $\omega(x)$,

$$\omega_j^k = \frac{1}{h_k}\omega(\frac{x - x_j^k}{h_k}).$$

Each k represents a different level of resolution of the function $f(x)$.

The set of values $\left\{\{\bar{f}_j^k\}_{j=1}^{J_k}\right\}_{k=0}^{L}$ is called a multiresolution analysis of $f(x)$, if for each k the knowledge of $\{\bar{f}_j^k\}_{j=1}^{J_k}$ determines the values on the next level $\{\bar{f}_j^{k-1}\}_{j=1}^{J_{k-1}}$. This means that the k-th level of resolution contains the information of all larger scales of variation.

It is proven in [11] that

$$\bar{f}_j^k = \sum_l \alpha_l \bar{f}_{2j+l}^{k+1} \tag{5}$$

is equivalent to a dilation relation for the weight function $\omega(x)$,

$$\omega(y) = 2\sum_l \alpha_l \omega(2y - l). \tag{6}$$

thus, if $\omega(x)$ satisfies a dilation relation, the averages $\{\bar{f}_j^k\}$ at the k-th level of resolution contain the information on all coarser grids and the set $\left\{\{\bar{f}_j^k\}_{j=1}^{J_k}\right\}_{k=0}^{L}$ constitutes a multiresolution analysis of $f(x)$.

Many of the functions $\omega(x)$ that are used in numerical analysis automatically satisfy a dilation equation. For example $\omega(x) = \delta(x)$, where δ is the Dirac distribution, satisfies

$$\omega(x) = 2\omega(2x) \Rightarrow \alpha_0 = 1; \tag{7}$$

$$\text{the box function} \quad \omega(x) = \begin{cases} 1 & -1 \le x < 0 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

satisfies

$$\omega(x) = \omega(2x) + \omega(2x + 1) \Rightarrow \alpha_0 = \alpha_{-1} = \frac{1}{2}; \tag{9}$$

$$\text{the hat function} \quad \omega(x) = \begin{cases} 1 + x & -1 \le x \le 0 \\ 1 - x & 0 \le x \le 1 \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

satisfies

$$\omega(x) = \frac{1}{2}[\omega(2x - 1) + 2\omega(2x) + \omega(2x + 1)] \Rightarrow \alpha_1 = \alpha_{-1} = \frac{1}{4}, \alpha_0 = \frac{1}{2}; \tag{11}$$

and the quadratic spline function

$$\omega(x) = \begin{cases} (x+2)^2 & -2 \le x \le -1 \\ -2x^2 - 2x + 1 & -1 \le x \le 0 \\ (x-1)^2 & 0 \le x \le 1 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

satisfies

$$\begin{aligned} \omega(x) &= \frac{1}{4}[\omega(2x-1) + 3\omega(2x) + 3\omega(2x+1) + \omega(2x+2)] \\ &\Rightarrow \quad \alpha_{-2} = \alpha_1 = \frac{1}{8}, \alpha_{-1} = \alpha_0 = \frac{3}{8}. \end{aligned} \tag{13}$$

All these functions $\omega(x)$ form a hierarchy of functions $\omega^m(x)$ which is obtained by repeated convolution with a characteristic function

$$\omega^{m+1} = \omega^m * \chi_{[-1+s_m, s_m]}, \qquad s_m = \frac{1}{2}[1 - (-1)^m], \tag{14}$$

with

$$\omega^0 = \delta(x) \tag{15}$$

Let α_l^m denote the coefficients of the dilation equation (6) which is satisfied by ω^m. It is easy to see that

$$\alpha_l^{m+1} = \frac{1}{2}(\alpha_l^m + \alpha_{l+(-1)^m}^m). \tag{16}$$

The shift between $\chi_{[-1,0]}$ and $\chi_{[0,1]}$ keeps the coefficients α_l^m as centered as possible around $l = 0$, which is convenient for formulating boundary conditions.

Discretizing via a dilation relation and on uniform grids is by no means an essential ingredient to the multi-resolution concept. The underlying idea of a multiresolution setting is that knowledge of the function on a scale determines the values of the discretization on all coarser scales.

Any linear operator, $\mathcal{D}$, which is defined on a space of functions $\mathcal{F}$ and takes values in a Eucledian vector space V of finite dimension J

$$\mathcal{D} : \mathcal{F} \longrightarrow V \qquad \dim V = J$$

can be considered as a discretization operator on the space of functions $\mathcal{F}$. We shall refer to J as the resolution of the discretization.

A sequence of discretization operators $\{\mathcal{D}_k\}$ with monotone increasing levels of resolution $\{J_k\}$, i.e.

$$\mathcal{D}_k : \mathcal{F} \longrightarrow V^k \qquad \dim V^k = J_k$$

$$\mathcal{D}_k f = \bar{f}^k \quad \in V^k \qquad \forall f \in \mathcal{F}$$

defines a multiresolution setting (see [13]) if there exists a matrix D_k^{k-1} such that

$$\text{rank } (D_k^{k-1}) = J_{k-1} \tag{17}$$

and

$$\bar{f}^{k-1} = D_k^{k-1} \bar{f}^k \tag{18}$$

The above relations mean that $\bar{f}^{k-1}$ can be evaluated from $\bar{f}^k$ without explicit knowledge of the function itself. Furthermore, it shows that if $\{\mathcal{D}_k\}$ is a multiresolution setting, then knowledge of the discretization of $f(x)$ for some level of resolution implies knowledge for all coarser levels.

In general, a given sequence of discretizations, $\{\mathcal{D}_k\}$, constitutes a multiresolution setting if (see [13])

- The sequence is nested, i.e.: $\mathcal{D}_k f = 0 \Rightarrow \mathcal{D}_{k-1} f = 0$
- Each $\mathcal{D}_k$ is reconstructible, i.e $\exists\ \mathcal{R}_k$: $\mathcal{D}_k \mathcal{R}_k = I$

where I is the identity operator in the corresponding finite dimensional space.

The decimation matrix, D_k^{k-1}, can then be defined as

$$D_k^{k-1} = \mathcal{D}_{k-1} \mathcal{R}_k \tag{19}$$

Observe that since $\bar{f}^{k-1}$ and $\bar{f}^k$ depend only on the discretization operators, the same is true for the matrix D_k^{k-1} in (18). It follows therefore that the matrix expression for D_k^{k-1} in (19) is the same for any $\mathcal{R}_k$ thus it does not depend at all on the reconstruction operator, but only on its existence.

In the case where the discretization operators, $\mathcal{D}_k$, are given by (4), D_k^{k-1} is a Toeplitz matrix

$$(D_k^{k-1})_{i,j} = \alpha_{j-2i}, \text{ independent of } k \tag{20}$$

but the framework is general enough to include discretizations corresponding to unstructured grids in several space-dimensions, as well as non-local discretizations (see [13] for details and examples).

3. Multiresolution representation. In the previous section we have made use of a reconstruction operator $\mathcal{R}_k$ that plays only an auxiliary role in characterizing a multiresolution setting. However, the multiresolution analysis allows for a decomposition of the original signal into scales and, here, the reconstruction operator plays a significant role.

A reconstruction procedure for a given discretization $\mathcal{D}$ is an operator $\mathcal{R}$ such that (see [13])

$$\mathcal{R} : V \longrightarrow \mathcal{F}$$

$$\mathcal{D}\mathcal{R} = I \tag{21}$$

If the sequence $\{\mathcal{D}_k\}$ constitutes a multiresolution setting, given $\bar{f}^{k-1} = \mathcal{D}_{k-1} f$ and using a reconstruction procedure at each level of resolution, we can get an approximation $\tilde{f}^k$ to $\bar{f}^k = \mathcal{D}_k f$ by

$$\tilde{f}^k = P_{k-1}^k \bar{f}^{k-1}, \quad P_{k-1}^k = \mathcal{D}_k \mathcal{R}_{k-1}. \tag{22}$$

We refer to P_{k-1}^k as the prediction operator and to

$$e^k = \bar{f}^k - \tilde{f}^k = \bar{f}^k - P_{k-1}^k \bar{f}^{k-1} \tag{23}$$

as the prediction error.

The prediction errors measure our success in using the reconstruction procedure to predict $\bar{f}^k$ from the knowledge of $\bar{f}^{k-1}$.

It is proven in [13] that

$$D_k^{k-1}\tilde{f}^k = \bar{f}^{k-1} = D_k^{k-1}\bar{f}^k. \tag{24}$$

Thus,

$$D_k^{k-1}e^k = 0. \tag{25}$$

Since

$$\operatorname{rank}(D_k^{k-1}) = J_{k-1}, \tag{26}$$

relation (25) is a homogeneous system of J_{k-1} linear equations for the J_k components of e^k. We conclude that e^k can be expressed in terms of $\Delta_k = J_k - J_{k-1}$ independent quantities, which we denote by the column-vector d^k.

In what follows, we shall consider only the dyadic case, thus $J_k = 2J_{k-1}$, and $\Delta_k = J_{k-1}$. When the discretization operators $\mathcal{D}_k$ come from a weight function that satisfies a dilation relation like (6), (25) becomes

$$\sum_l \alpha_l e_{2j+l}^k = 0 \tag{27}$$

For $0 \leq m \leq 7$, the weight functions in the hierarchy (14) satisfy dilation relations whose coefficients have the following property

$$|\alpha_0| > \sum_{l\neq 0} |\alpha_l|, \tag{28}$$

hence, since only half of the prediction errors are independent, it is possible to store the values e_j^k with odd indices and use the relation (27) in order to formulate a system of equations

$$\sum \alpha_{2l} e_{2j+2l}^k = -\sum \alpha_{2l-1} e_{2j+2l-1}^k \tag{29}$$

for the unknowns $(e_2^k, e_4^k, \ldots, e_{J_k}^k)$. Condition (28) implies that the matrix of the system (29) is strictly diagonally dominant and hence invertible.

Let us denote by G_k^{k-1} and T_{k-1}^k the matrices that transfer $\{e_j^k\}$ into $\{d_j^k\}$ and vice versa; i.e.

$$d^k = G_k^{k-1}e^k \tag{30}$$

$$e^k = T_{k-1}^k d^k \tag{31}$$

In the dilation relation case, we can use the same transfer matrix, G_k^{k-1}, for all weight functions. Algorithmically, it can be expressed as

$$d_j^k = e_{2j-1}^k \qquad 1 \leq j \leq J_{k-1} \tag{32}$$

i.e. $(G_k^{k-1})_{ij} = \delta_{2i-1,j}$.

For the first three members of the hierarchy (14), T_{k-1}^k can be expressed algorithmically

as follows,

$$\omega(x) = \delta(x) \qquad \begin{cases} e^k_{2j-1} &= d^k_j \\ e^k_{2j} &= 0 \end{cases} \qquad 1 \le j \le J_{k-1} \tag{33}$$

$$\omega(x) = \text{ box function} \qquad \begin{cases} e^k_{2j-1} &= d^k_j \\ e^k_{2j} &= -d^k_j \end{cases} \qquad 1 \le j \le J_{k-1} \tag{34}$$

$$\omega(x) = \text{ hat function} \qquad \begin{cases} e^k_{2j-1} &= d^k_j \\ e^k_{2j} &= -\frac{1}{2}(d^k_j + d^k_{j+1}) \end{cases} \qquad 1 \le j \le J_{k-1} \tag{35}$$

In the general framework, the multiresolution analysis need not have a direct relation to any dilation equation. In this case, one can also construct the transformations between e^k and d^k as a generalization of the ones which are used for orthogonal wavelets.

It is shown in [13] that, in this case

$$d^k = G^{k-1}_k e^k \tag{36}$$

with G^{k-1}_k a $J_{k-1} \times J_k$ matrix whose rank is J_{k-1} and such that

$$D^{k-1}_k (G^k_{k-1})^* = 0. \tag{37}$$

The inverse transformation is as follows,

$$e^k = T^k_{k-1} d^k = S^{-1}_k (G^{k-1}_k)^* d^k \tag{38}$$

where

$$S_k = (D^{k-1}_k)^* D^{k-1}_k + (G^{k-1}_k)^* G^{k-1}_k \tag{39}$$

we refer to [13] for a precise definition of G^{k-1}_k.

From (23) we have

$$\bar{f}^k = P^k_{k-1} \bar{f}^{k-1} + e^k = P^k_{k-1} \bar{f}^{k-1} + T^k_{k-1} d^k. \tag{40}$$

Observe that the number of elements on $\{\bar{f}^k_j\}_{j=1}^{J_k}$ is equal to the number of elements on $\{\bar{f}^{k-1}_j\}_{j=1}^{J_{k-1}}$ plus the number of elements of $\{d^k_j\}_{j=1}^{J_{k-1}}$. We can interpret relation (40) as saying that d^k represents non redundant information which is present on $\bar{f}^k$ and is not predictable from $\bar{f}^{k-1}$ by the reconstruction procedure $\mathcal{R}_k$. Motivated by this interpretation, the components of d^k are referred to as the k-th scale coefficients of the multiresolution sequence.

Given $\bar{f}^L \in V^L$, we define the column vector

$$\mu(\bar{f}^L) = \begin{pmatrix} d^L \\ \vdots \\ d^1 \\ \bar{f}^0 \end{pmatrix} \in V^L. \tag{41}$$

It is shown in [13] that if the sequence $\{(\mathcal{D}_k, \mathcal{R}_k)\}$ satisfies (21) and $\{\mathcal{D}_k\}$ defines a

multiresolution setting, there is a one-to-one transformation between $\bar{f}^L$ and $\mu(\bar{f}^L)$, M, such that

$$\mu(\bar{f}^L) = M\bar{f}^L; \quad \bar{f}^L = M^{-1}\mu(\bar{f}^L). \tag{42}$$

The algorithms to carry out this transformation and its inverse are as follows:

$\mu(\bar{f}^L) = M\bar{f}^L$ (Encoding)

$$\left\{ \begin{array}{ll} \text{Do} & k = L, 1 \\ & \bar{f}^{k-1} = D_k^{k-1}\bar{f}^k \\ & d^k = G_k^{k-1}(\bar{f}^k - P_{k-1}^k\bar{f}^{k-1}) \end{array} \right. \tag{43}$$

$\bar{f}^L = M^{-1}\mu(\bar{f}^L)$ (Decoding)

$$\left\{ \begin{array}{ll} \text{Do} & k = 1, L \\ & \bar{f}^k = P_{k-1}^k\bar{f}^{k-1} + T_{k-1}^k d^k \end{array} \right. \tag{44}$$

4. Interpolatory and Cell-average Multiresolution . All members of the hierarchy (14) lead to discrete multi-resolution analysis with discretizations on the sequence of nested dyadic grids $\{X^k\}_{k=0}^{+\infty}$

$$X^k = \{x_j^k\}_{j=0}^{J_k},\ x_j^k = j.h_k,\ h_k = 2^{-k}h_0,\ J_k = \frac{1}{h_k} = 2^k J_0 \tag{45}$$

The multi-resolution settings derived from the first two members of the hierarchy, Dirac's delta distribution and the box function, have been described and analyzed in [11, 13].

Dirac's delta function gives rise to interpolatory multi-resolution settings. These are appropriate for multiscale representations of continuous functions. We highlight its main properties here and refer the reader to [11, 13] for a more complete development.

$$\mathcal{D}_k : \mathcal{C}[0,1] \longrightarrow V^k \tag{46}$$

$$\bar{f}_j^k = (\mathcal{D}_k f)_j = f(x_j^k), \quad 0 \leq j \leq J_k \tag{47}$$

$$D_k^{k-1} = \delta_{2i,j} \qquad G_k^{k-1} = \delta_{2i-1,j} \tag{48}$$

The reconstruction procedure for this discretization is any operator $\mathcal{R}_k$ such that

$$\mathcal{R}_k : V^k \longrightarrow \mathcal{C}[0,1] \tag{49}$$

$$\mathcal{D}_k\mathcal{R}_k\bar{f}^k = \bar{f}^k \tag{50}$$

i.e

$$(\mathcal{R}_k\bar{f}^k)(x_j^k) = \bar{f}_j^k = f(x_j^k); \tag{51}$$

that is, $\mathcal{R}_k$ is any continuous interpolation of the data $\bar{f}^k$ at the grid points of X^k.

If we denote

$$(\mathcal{R}_k \bar{f}^k)(x) = I_k(x; \bar{f}^k)$$

the encoding and decoding algorithms are

$\mu(\bar{f}^L) = M\bar{f}^L$ (Encoding)

$$\text{(52)} \qquad \begin{cases} \text{Do} \quad k = L, 1 \\ \qquad \bar{f}_j^{k-1} = \bar{f}_{2j}^k, & 1 \le j \le J_{k-1} \\ \qquad d_j^k = \bar{f}_{2j-1}^k - I_{k-1}(x_{2j-1}^k; \bar{f}^k), & 1 \le j \le J_{k-1} \end{cases}$$

$\bar{f}^L = M^{-1}\mu(\bar{f}^L)$ (Decoding)

$$\text{(53)} \qquad \begin{cases} \text{Do} \quad k = 1, L \\ \qquad \bar{f}_{2j}^k = \bar{f}_j^{k-1}, & 1 \le j \le J_{k-1} \\ \qquad \bar{f}_{2j-1}^k = I_{k-1}(x_{2j-1}^k; \bar{f}^{k-1}) + d_j^k, & 1 \le j \le J_{k-1} \end{cases}$$

The second member of the hierarchy, the box function, leads to the so-called cell-average multi-resolution. Now we have

$$\text{(54)} \qquad \mathcal{D}_k : L^1[0,1] \longrightarrow V^k$$

$$\text{(55)} \qquad \bar{f}_j^k = (\mathcal{D}_k f)_j = \frac{1}{h_k} \int_{x_{j-1}^k}^{x_j^k} f(x)dx, \quad 1 \le j \le J_k;$$

where $L^1[0,1]$ is the space of absolutely integrable functions in $[0,1]$. This analysis turns out to be appropriate for data compression of discontinuous, piecewise smooth signals.

In this case

$$\text{(56)} \qquad (D_k^{k-1})_{i,j} = \frac{1}{2}(\delta_{2i-1,j} + \delta_{2i,j})$$

and

$$\text{(57)} \qquad (G_k^{k-1})_{i,j} = \frac{1}{2}(\delta_{2i-1,j} - \delta_{2i,j})$$

the reconstruction operator $\mathcal{R}_k$ for this discretization is any operator

$$\mathcal{R}_k : V_k \longrightarrow L^1[0,1]$$

that satisfies

$$\text{(58)} \qquad (\mathcal{D}_k \mathcal{R}_k \bar{f}^k)_j = \frac{1}{h_k} \int_{x_{j-1}^k}^{x_j^k} (\mathcal{R}_k \bar{f}^k)(x)dx = \bar{f}_j^k$$

The simplest way to construct $\mathcal{R}_k$ is via its 'primitive function'. Let us define the

sequence $\{F_j^k\}$ on the k-th grid as

$$F_j^k = h_k \sum_{i=1}^{j} \bar{f}_j^k = F(x_j^k) = \int_0^{x_j^k} f(x)dx.$$

Thus, $F(x)$ is the primitive of $f(x)$ and the sequence $\{F_j^k\}$ corresponds to a discretization by point-values of $F(x)$ on the k-th grid. Let us denote by $I_k(x, F^k)$ an interpolatory reconstruction of $F(x)$, and define

$$(\mathcal{R}_k \bar{f}^k)(x) = \frac{d}{dx} I_k(x; F^k).$$

Clearly $(\mathcal{R}_k \bar{f}^k)(x) \in L^1([0,1])$, and it is easy to see that $\mathcal{D}_k \mathcal{R}_k = I_k$.

The multi-resolution transform and its inverse are now

$\mu(\bar{f}^L) = M\bar{f}^L$ (Encoding)

$$\text{(59)} \qquad \begin{cases} \text{Do} \quad k = L, 1 \\ \qquad \bar{f}_i^{k-1} = \frac{1}{2}(\bar{f}_{2i-1}^k + \bar{f}_{2i}^k), & 1 \le i \le J_{k-1} \\ \qquad d_i^k = \frac{1}{h_k}(F_{2i-1}^k - I_{k-1}(x_{2i-1}^k; F^{k-1})), & 1 \le i \le J_{k-1} \end{cases}$$

$\bar{f}^L = M^{-1}\mu(\bar{f}^L)$ (Decoding)

$$\text{(60)} \qquad \begin{cases} \text{Do} \quad k = 1, L \\ \qquad \bar{f}_{2i-1}^k = \frac{1}{h_k}(I_{k-1}(x_{2i-1}^k; F^{k-1}) - F_{i-1}^{k-1}) + d_i^k, & 1 \le i \le J_{k-1} \\ \qquad \bar{f}_{2i}^k = 2\bar{f}_i^{k-1} - \bar{f}_{2i-1}^k, & 1 \le i \le J_{k-1} \end{cases}$$

The next member in the hierarchy is the hat function. In the next section, we analyze in detail the properties of the multi-resolution setting it defines, and characterize the reconstruction operators as well as the natural function space for the discrete approximations.

5. Hat-weighted Multiresolution . We define the discretization operators as follows:

$$\text{(61)} \qquad \mathcal{D}_k : \mathcal{F} \longrightarrow V_k$$

$$(\mathcal{D}_k f)_j = \bar{f}_j^k = < f, \omega_j^k >$$

where

$$\omega_j^k = \frac{1}{h_k}\omega(\frac{x}{h_k} - j)$$

and $\omega(x)$ is the hat function (10). Hence

$$\text{(62)} \qquad \bar{f}_j^k = \frac{1}{h_k}\int_{x_{j-1}^k}^{x_j^k} f(x)(1 + \frac{x - x_j^k}{h_k})dx + \frac{1}{h_k}\int_{x_j^k}^{x_{j+1}^k} f(x)(1 - \frac{x - x_j^k}{h_k})dx.$$

Notice that there is an essential difference between this framework and those described

in the previous section already at the level of the discretizations. In order to compute $\bar{f}^k_{J_k}$ we need to know the function in $[x^k_{J_k}, x^k_{J_k+1}] = [1, 1+h_k]$. In this paper, we assume that $f(x)$ is a periodic function. In the non periodic case, one has to extrapolate $f(x)$ outside its domain of definition .

From the dilation relation (11) or (in the spirit of [13]) directly from (62), we obtain

$$\bar{f}^{k-1}_j = \frac{1}{4}\bar{f}^k_{2j-1} + \frac{1}{2}\bar{f}^k_{2j} + \frac{1}{4}\bar{f}^k_{2j+1}.$$

thus the decimation matrix D^{k-1}_k is given by

$$(D^{k-1}_k)_{i,j} = \frac{1}{4}\delta_{2i-1,j} + \frac{1}{2}\delta_{2i,j} + \frac{1}{4}\delta_{2i+1,j} \tag{63}$$

Since we are assuming periodicity, the matrix in (63) has to be interpreted in a cyclic manner. Note that $\mathrm{rank}(D^{k-1}_k) = J_{k-1}$.

The reconstruction procedures $\mathcal{R}_k$ are operators

$$\mathcal{R}_k : V_k \longrightarrow \mathcal{F} \tag{64}$$

which satisfy

$$(\mathcal{D}_k \mathcal{R}_k \bar{f}^k)_j = \bar{f}^k_j, \quad 1 \le j \le J_k. \tag{65}$$

Let us describe a procedure to compute the reconstruction operators for the hat-averaged multi-resolution analysis which is a generalization of the 'reconstruction via primitive function' developed in the context of cell averages. We refer to this procedure as reconstruction via "second primitive".

We start by proving the following lemma:

LEMMA 5.1. *Let $f(x)$ be a periodic integrable function and let $H(x)$ be a continuous, piecewise smooth function such that $H''(x) = f(x)$ almost everywere in $[x^k_{j-1}, x^k_j]$ for each $j = 1, \cdots J_k + 1$. Then*

$$\bar{f}^k_j = \frac{1}{h_k}\left[H'_-(x^k_j) - H'_+(x^k_j)\right] + \frac{1}{h_k^2}(H^k_{j+1} - 2H^k_j + H^k_{j-1}), \quad 1 \le j \le J_k \tag{66}$$

Proof.

$$\begin{aligned}
\bar{f}^k_j &= \frac{1}{h_k}\int_{x^k_j}^{x^k_{j-1}} f(x)(1 + \frac{x - x^k_j}{h_k})dx + \frac{1}{h_k}\int_{x^k_j}^{x^k_{j+1}} f(x)(1 - \frac{x - x^k_j}{h_k})dx \\
&= \frac{1}{h_k}\int_{x^k_j}^{x^k_{j-1}} H''(x)(1 + \frac{x - x^k_j}{h_k})dx + \frac{1}{h_k}\int_{x^k_j}^{x^k_{j+1}} H''(x)(1 - \frac{x - x^k_j}{h_k})dx
\end{aligned}$$

and integrate by parts. □

Consider the function

$$H(x) = \int_0^x \int_0^y f(z)dzdy, \qquad H^k_j = H(x^k_j), \quad 1 \le j \le J_k + 1. \tag{67}$$

This is a $\mathcal{C}^2$ function that satisfies the lemma, thus

$$\bar{f}_j^k = \frac{1}{h_k^2}(H_{j+1}^k - 2H_j^k + H_{j-1}^k), \quad 1 \le j \le J_k. \tag{68}$$

The function 67 is one of the "second primitives" of $f(x)$. A more convenient one is

$$H(x) = \int_0^x \int_0^y f(z)dzdy - \alpha x \tag{69}$$

$$\alpha = \int_0^1 \int_0^y f(z)dzdy \tag{70}$$

For this particular "second primitive" $H_0^k = H(0) = H_{J_k}^k = H(1) = 0$, and it is possible to establish a one-to-one correspondence between the sets

$$\{\bar{f}_j^k\}_{j=1}^{J_k} \qquad \text{and} \qquad \{H_j^k\}_{j=1}^{J_k+1} \tag{71}$$

In fact, from (68) it is easy to verify the following relations

$$jH_{j+1}^k - (j+1)H_j^k = h_k^2\left(\bar{f}_1^k + 2\bar{f}_2^k + \cdots + j\bar{f}_j^k\right) \qquad 1 \le j \le J_k. \tag{72}$$

and, since $H_0^k = H_{J_k}^k = 0$, it is evident that the knowledge of the set $\{\bar{f}_j^k\}_{j=1}^{J_k-1}$ is equivalent to the knowledge of the set $\{H_j^k\}_{j=1}^{J_k-1}$. $\bar{f}_{J_k}^k$ is related to values of $H(x)$ outside of the interval $[0,1]$.

The observations above immediately suggest the following reconstruction technique: Interpolate the point-values of the "second primitive" by any interpolation procedure $I_k(x; H^k)$ and define

$$(\mathcal{R}_k \bar{f}^k)(x) = \frac{d^2}{dx^2} I_k(x; H^k) \tag{73}$$

In general, $I_k(x; H^k)$ is a continuous, piecewise smooth function. Its first derivative will also be a piecewise smooth function possibly with discontinuities at the grid points of the k-th level, thus its second derivative must be considered in the sense of distributions. An appropriate definition for $\mathcal{F}$ is, thus, the space of distributions that can be represented as piecewise smooth functions with delta function sigularities at certain locations of their domain of definition.

To prove (65) we shall use the following lemma, whose proof is a straight application of the definition of distributional derivative and shall be omitted.

LEMMA 5.2. *Let $I(x)$ be a piecewise smooth function of the form*

$$I(x) = \begin{cases} I_L(x) & \text{if } x < 0 \\ I_R(x) & \text{if } x > 0. \end{cases} \tag{74}$$

Then, its derivative in the distribution sense is

$$\frac{d}{dx} I(x) = \tilde{I}(x) + (I_R(0) - I_L(0))\delta(x) \tag{75}$$

where

$$\tilde{I}(x) = \begin{cases} \frac{d}{dx} I_L(x) & \text{if } x < 0 \\ \frac{d}{dx} I_R(x) & \text{if } x > 0. \end{cases} \tag{76}$$

Therefore, if the interpolatory function is defined as

$$I_k(x; H^k) = I_{k,j}(x) \qquad \text{for } x \in [x^k_{j-1}, x^k_j] \quad 1 \le j \le J_k + 1 \tag{77}$$

and

$$(\mathcal{R}_k \bar{f}^k)(x) = \frac{d^2}{dx^2} I_k(x; H^k) \tag{78}$$

in the distribution sense, we have that

$$(\mathcal{R}_k \bar{f}^k)(x) = \tilde{I}_k(x) + \sum_{j=1}^{J_k} s^k_j \delta(x - x^k_j) \qquad x \in [0,1] \tag{79}$$

where $\tilde{I}_k$ is defined as

$$\tilde{I}_k(x) = \frac{d^2}{dx^2} I_{k,j}(x) \qquad \text{for } x \in [x^k_{j-1}, x^k_j], \quad 1 \le j \le J_k + 1 \tag{80}$$

and

$$s^k_j = \left[\frac{d}{dx} I_{k,j+1}(x) - \frac{d}{dx} I_{k,j}(x) \right]_{x = x^k_j} \tag{81}$$

Let us now prove (65);

$$\begin{aligned}
(\mathcal{D}_k \mathcal{R}_k \bar{f}^k)_j &= < \mathcal{R}_k \bar{f}^k, \omega^k_j > = < \tilde{I}(x) + \sum_{l=1}^{J_k} \delta(x - x^k_l) s^k_l, \omega^k_j > \\
&= < \tilde{I}(x), \omega^k_j > + \sum_{l=1}^{J_k} s^k_l < \delta(x - x^k_l), \omega^k_j > \\
&= \frac{1}{h_k} \int_{x^k_{j-1}}^{x^k_j} \frac{d^2}{dx^2} I_{k,j}(x)(1 + \frac{x - x^k_j}{h_k}) dx + \\
&\quad \frac{1}{h_k} \int_{x^k_j}^{x^k_{j+1}} \frac{d^2}{dx^2} I_{k,j+1}(x)(1 - \frac{x - x^k_j}{h_k}) dx + s^k_j \frac{1}{h_k} \\
&= \frac{1}{h_k} (\frac{d}{dx} I_{k,j})(x^k_j) - \frac{1}{h_k^2} \left[I_{k,j}(x^k_j) - I_{k,j}(x^k_{j-1}) \right] - \\
&\quad \frac{1}{h_k} (\frac{d}{dx} I_{k,j+1})(x^k_j) + \frac{1}{h_k^2} \left[I_{k,j+1}(x^k_{j+1}) - I_{k,j+1}(x^k_j) \right] + s^k_j \frac{1}{h_k} \\
&= \frac{1}{h_k^2} [H^k_{j+1} - 2H^k_j + H^k_{j-1}] = \bar{f}^k_j
\end{aligned}$$

The predicted values $\tilde{f}^k$ are computed as follows:

$$\begin{aligned}
\tilde{f}^k_{2j-1} &= (\mathcal{D}_k\mathcal{R}_{k-1}\bar{f}^{k-1})_j = <\mathcal{R}_{k-1}\bar{f}^{k-1},\omega^k_{2j-1}> \\
&= <\tilde{I}_{k-1},\omega^k_{2j-1}> + \sum_{l=1}^{J_{k-1}} <\delta(x-x^{k-1}_j),\omega^k_{2j-1}> \\
&= \frac{1}{h_k}\int_{x^k_{2j-2}}^{x^k_{2j-1}} \frac{d^2}{dx^2}I_{k-1,j}(x)(1+\frac{x-x^k_{2j-1}}{h_k})dx + \\
&\quad \frac{1}{h_k}\int_{x^k_{2j-1}}^{x^k_{2j}} \frac{d^2}{dx^2}I_{k-1,j}(x)(1-\frac{x-x^k_{2j-1}}{h_k})dx \\
&= \frac{1}{h_k^2}\left(I_{k-1,j}(x^k_{2j-2};H^{k-1}) - 2I_{k-1,j}(x^k_{2j-1};H^{k-1}) + I_{k-1,j}(x^k_{2j};H^{k-1})\right)
\end{aligned}$$

Analogously,

$$\begin{aligned}
\tilde{f}^k_{2j} &= (\mathcal{D}_k\mathcal{R}_{k-1}\bar{f}^{k-1})_j = <\mathcal{R}_{k-1}\bar{f}^{k-1},\omega^k_{2j}> \\
&= \frac{1}{h_k^2}\left(I_{k-1,j}(x^k_{2j-1};H^{k-1}) - 2I_{k-1,j}(x^k_{2j};H^{k-1}) + I_{k-1,j+1}(x^k_{2j+1};H^{k-1})\right)
\end{aligned}$$

Thus,

$$\begin{aligned}
\tilde{f}^k_j &= <\mathcal{R}_{k-1}\bar{f}^{k-1},\omega^k_j> \\
&= \frac{1}{h_k^2}\left(I_{k-1}(x^k_{j-1};H^{k-1}) - 2I_{k-1}(x^k_j;H^{k-1}) + I_{k-1}(x^k_{j+1};H^{k-1})\right)
\end{aligned} \tag{82}$$

Since $f(x)$ is periodic, and $H(0) = H(1) = 0$ we would like to extend (69) periodically outside the unit interval. This would also avoid the 'awkward' choice of H_{J_k+1} as a base point for the interpolation. However, lemma 5.1 would then imply that

$$\begin{aligned}
\bar{f}^k_{J_k} &= \frac{1}{h_k}[H'(1)-H'(0)] + \frac{1}{h_k^2}(H^k_{J_k+1} - 2H^k_{J_k} + H^k_{J_k-1}) \\
&= \frac{1}{h_k}\int_0^1 f(x)dx + \frac{1}{h_k^2}(H^k_1 + H^k_{J_k-1})
\end{aligned} \tag{83}$$

since this periodic extension would not, in general, be a $\mathcal{C}^1$ function.

Since

$$\int_1^{1+h_k} f(x)(1-\frac{x-x^k_{J_k}}{h_k})dx = \int_0^{h_k} f(x)(1-\frac{x-x^k_0}{h_k})dx$$

when $f(x)$ is periodic, it is easy to prove that

$$h_k\sum_{j=1}^{J_k}\bar{f}^k_j = \int_0^1 f(x)dx \tag{84}$$

Equations (68) and (83) imply that, $H_{J_k+1} = H(1+h_k) = H_1 + h_k\int_0^1 f(x)dx$, i.e. the sets $\{\bar{f}^k_j\}_{j=1}^{J_k}$ and $\{H^k_j\}_{j=1}^{J_k-1}\bigcup\{\int_0^1 f(x)dx\}$ are also equivalent in the sense that knowledge of one implies knowledge of the other and viceversa. It is then advantageous to work with functions which have zero mean. Instead of the discrete set $\{\bar{f}^k_j\}$ we consider the set $\{\bar{f}^k_j - h_k\sum\bar{f}^k_j\}$. These represent the hat-averages of a function with zero average.

Without loss of generality, we can assume that $f(x)$ is a function such that $\int_0^1 f(x)dx = 0$ and consider its second primitive (69) extended periodically outside the interval $[0,1]$. Then

$$(\mathcal{R}_k \bar{f}^k)(x) = \frac{d^2}{dx^2} I_k(x; H^k)$$

is given by (79),(80),(81) with

$$I_{k,J_k+1}(x) = I_{k,1}(x)$$

Notice that, since

$$I_{k-1}(x_{2i}^k; H^{k-1}) = I_{k-1}(x_i^{k-1}; H^{k-1}) = H_i^{k-1} = H_{2i}^k,$$

we have that

$$\begin{aligned}
(85) \quad e_{2j-1}^k(f) &= \bar{f}_{2j-1}^k - \tilde{f}_{2j-1} \\
&= \frac{1}{h_k^2}[H_{2j}^k - 2H_{2j-1}^k + H_{2j-2}^k] - \\
&\quad \frac{1}{h_k^2}\left(I_j^{k-1}(x_{2j-2}^k; H^{k-1}) - 2I_j^{k-1}(x_{2j-1}^k; H^{k-1}) + I_j^{k-1}(x_{2j}^k; H^{k-1})\right) \\
(86) \quad &= -\frac{2}{h_k^2}(H_{2j-1}^k - I_j^{k-1}(x_{2j-1}^k; H^{k-1})) = -\frac{2}{h_k^2} e_{2j-1}^k(H)
\end{aligned}$$

thus, with the periodic choice, $e_{J_k+1}^k(f) = e_1^k(f)$, i.e. the prediction errors for f are also periodic.

As we have pointed out in section 3, the relation $D_k^{k-1} e^k = 0$ means that the prediction errors at the k-th level can be expressed in terms of J_{k-1} independent quantities, the k-th scale coefficients d^k.

Since the discretization operators in (62) come from taking the hat function as the weight function, we can define the transfer matrix G_k^{k-1} as in (32). We refer to this choice as *non-orthogonal.*

On the other hand, we can also define the transfer matrices in a wavelet-like manner, as in (37) and (38). We refer to this choice as *orthogonal.*

It is interesting to notice that both choices lead to the same algorithms and the same scale coefficients for the interpolatory and cell-average multiresolution transforms, but this is not so for the hat-average case.

Let us describe the orthogonal and non-orthogonal choices for the hat-averaged multiresolution.

5.1. Non-Orthogonal case.

We rewrite relation $D_k^{k-1} e^k = 0$ as

$$(87) \qquad e_{2i}^k = -\frac{1}{2} e_{2i-1}^k - \frac{1}{2} e_{2i+1}^k, \quad 1 \le i \le J_{k-1},$$

hence we have enough information with the odd indices of e^k, i.e. with $d^k := G_k^{k-1} e^k$, where G_k^{k-1} is the $J_{k-1} \times J_k$-matrix

$$(88) \qquad (G_k^{k-1})_{i,j} = \delta_{2i-1,j}.$$

We recover e^k with the algorithm:

$$\left\{ \begin{array}{lcl} e^k_{2j-1} & = & d^k_j \\ e^k_{2j} & = & -\frac{1}{2}(d^k_j + d^k_{j+1}) \end{array} \right. 1 \le j \le J_{k-1} \tag{89}$$

That is,

$$(T^k_{k-1})_{ij} = \left\{ \begin{array}{ll} \delta_{i-(2j-1)} & \text{if } i \text{ is odd} \\ -\frac{1}{2}(\delta_{i-2j} + \delta_{i-2(j+1)}) & \text{if } i \text{ is even} \end{array} \right.$$

5.2. Orthogonal case.

This case corresponds to the choice of G^{k-1}_k that mimics the wavelet framework, which is accomplished by taking (see [13])

$$(G^{k-1}_k)_{ij} = (-1)^{j+1}\alpha_{2i-j-1}$$

which in this case is just

$$(G^{k-1}_k)_{i,j} = -\frac{1}{4}\delta_{2i-2,j} + \frac{1}{2}\delta_{2i-1,j} - \frac{1}{4}\delta_{2i,j}. \tag{90}$$

It is easy to check that

$$\text{rank}(G^{k-1}_k) = J_{k-1}$$

and

$$D^{k-1}_k (G^{k-1}_k)^* = 0.$$

S_k in (39) is the $J_k \times J_k$ banded matrix:

$$(S_k)_{i,j} = \frac{1}{16}(\delta_{i-2,j} + 6\delta_{i,j} + \delta_{i+2,j}). \tag{91}$$

To recover e^k, from $d^k = G^{k-1}_k e^k$, we have to solve the system of linear equations:

$$S_k e^k = (G^{k-1}_k)^* d^k \tag{92}$$

which decouples into two separate systems for the odd and even components of e^k, i.e.

$$\frac{1}{16}e^k_{2(i+1)-1} + \frac{3}{8}e^k_{2i-1} + \frac{1}{16}e^k_{2(i-1)-1} = \frac{1}{2}d^k_i, \quad 1 \le i \le J_{k-1} \tag{93}$$

$$\frac{1}{16}e^k_{2(i+1)} + \frac{3}{8}e^k_{2i} + \frac{1}{16}e^k_{2(i-1)} = -\frac{1}{4}d^k_i - \frac{1}{4}d^k_{i+1}, \quad 1 \le i \le J_{k-1} \tag{94}$$

Let e^{odd} and e^{even} denote the column-vectors of the J_{k-1} odd and even components of e^k, respectively, and let $\hat{e}^{odd}$ and $\hat{e}^{even}$ denote the RHS of (93) and (94), respectively. We rewrite (93) and (94) in matrix form

$$\hat{S}e^{odd} = \hat{e}^{odd}, \hat{S}e^{even} = \hat{e}^{even}, \tag{95}$$

where $\hat{S}$ is given by (91).

These systems can be solved in $O(J_{k-1})$ operations using the LU decomposition of $\hat{S}$.

An explicit derivation is given in Appendix A.

Unless otherwise stated, in the remainder of the paper we compute the scale coefficients with the non-orthogonal option. The multiresolution transform and its inverse, are:

$\mu(\bar{f}^L) = M\bar{f}^L$ (Encoding)

$$(96)\quad \begin{cases} \text{Do} \quad k = L, 1 & \\ \quad \bar{f}_i^{k-1} = \frac{1}{4}(\bar{f}_{2i-1}^k + 2\bar{f}_{2i}^k + \bar{f}_{2i+1}^k), & 1 \le i \le J_{k-1} \\ \quad d_i^k = \bar{f}_{2i-1}^k - \tilde{f}_{2i-1}^k & \\ \quad = -\frac{2}{h_k^2}(H_{2i-1}^k - I_i^{k-1}(x_{2i-1}^k; H^{k-1})), & 1 \le i \le J_{k-1} \end{cases}$$

$\bar{f}^L = M^{-1}\mu(\bar{f}^L)$ (Decoding)

$$(97)\quad \begin{cases} \text{Do} \quad k = 1, L & \\ \quad \bar{f}_{2i-1}^k = \tilde{f}_{2i-1}^k + d_i^k & \\ \quad = \frac{1}{h_k^2}(H_i^{k-1} - H_{i-1}^{k-1} - 2I^{k-1}(x_{2i-1}^k; H^{k-1})), & 1 \le i \le J_{k-1} \\ \quad \bar{f}_{2i}^k = 2\bar{f}_i^{k-1} - \frac{1}{2}(\bar{f}_{2i-1}^k + \bar{f}_{2i+1}^k), & 1 \le i \le J_{k-1} \end{cases}$$

Up to this point, we have not specified the kind of interpolatory procedure to be used. The choice of a particular type of procedure depends on the particular application at hand.

Data independent interpolatory techniques lead to reconstruction operators that behave as linear functionals. In this case, the multi-resolution transform has an additional functional structure. On the other hand, data dependent interpolatory techniques lead to reconstruction operators which are non linear. These turn out to be appropriate for maximal compression of 'singular' signals.

We proceed now to study various examples of data-dependent and data-independent interpolatory procedures, the reconstruction operators that are derived from them, and the special properties of the multi-resolution representations obtained in each case.

6. Linear reconstruction operators and multi-resolution bases. When the reconstruction operators, $\mathcal{R}_k$ are linear functionals, the prediction operators P_k^{k+1} are matrices and the multi-resolution transform is a linear operator which describes a change of basis vectors.

Following [13], let us denote

$$A_m^L = \prod_{k=m}^{L-1} P_k^{k+1} = P_{L-1}^L \dots P_m^{m+1}, \qquad A_m^m = I;$$

it is easy to prove that

$$(98)\qquad \bar{f}^L = \sum_{m=1}^{L} A_m^L T_{m-1}^m d^m + A_0^L \bar{f}^0.$$

With the following definitions:

$$(\delta_i^J)_j = \delta_{ij}, \quad 1 \le i \le J$$

$$\eta_i^k = T_{k-1}^k \delta_i^{J_{k-1}}, \quad 1 \le i \le J_{k-1} \quad (\eta_i^k \in V^k)$$

$$\bar{\varphi}_i^{m,L} = A_m^L \delta_i^{J_m} \in V^L \qquad \bar{\psi}_i^{m,L} = A_m^L \eta_i^m = \sum_{j=1}^{J_m} (\eta_i^m)_j \bar{\varphi}_j^{m,L} \in V^L$$

we can rewrite (98) as

$$\bar{f}^L = \sum_{m=1}^{L} \sum_{i=1}^{J_{m-1}} d_i^m \bar{\psi}_i^{m,L} + \sum_{i=0}^{J_0} \bar{f}_i^0 \bar{\varphi}_i^{0,L}. \tag{99}$$

Thus,

$$\mathcal{B}_M = (\{\{\bar{\psi}_i^{m,L}\}_{i=1}^{J_{m-1}}\}_{m=1}^{L}, \{\bar{\varphi}_i^{0,L}\}_{i=1}^{J_0})$$

is a basis in V^L such that the coordinates of any $\bar{f}^L \in V^L$ in this basis are the components of the multi-resolution representation $\mu(\bar{f}^L)$. These coordinates can be computed by algorithm (43).

On the functional side, let us denote

$$\varphi_i^{m,L} = \mathcal{R}_L \bar{\varphi}_i^{m,L}; \quad \psi_i^{m,L} = \mathcal{R}_L \bar{\psi}_i^{m,L}$$

thus, $\varphi_i^{m,L}, \psi_i^{m,L} \in \mathcal{F}$. Using the linearity of the discretization operator, it follows immediately from (99) that for any $f \in \mathcal{F}$

$$(\mathcal{R}_L \mathcal{D}_L f)(x) = \sum_{m=1}^{L} \sum_{i=1}^{J_{m-1}} d_i^m(f) \psi_i^{m,L}(x) + \sum_{i=0}^{J_0} (\mathcal{D}_0 f)_i \varphi_i^{0,L}(x). \tag{100}$$

The multiresolution representation (100) may also correspond to a multi-scale decomposition. In order for this to happen, a sufficient condition is that

$$\exists \lim_{L \to \infty} \varphi_i^{m,L} =: \varphi_i^m \in \mathcal{F} \tag{101}$$

If (101) holds, then we have the following

$$(\hat{\mathcal{R}}_k \bar{f}^k)(x) = (\hat{\mathcal{R}}_0 \bar{f}^0)(x) + \sum_{k=1}^{L} \sum_{i=1}^{J_{k-1}} d_i^k(f) \psi_i^k(x) \tag{102}$$

where

$$\hat{\mathcal{R}}_k \bar{f}^k(x) = \sum_{i=1}^{J_k} \bar{f}_i^k \varphi_i^k \tag{103}$$

and

$$\psi_i^m = \lim_{L \to +\infty} \psi_i^{m,L} \tag{104}$$

The sequence $\{(\mathcal{D}_k, \hat{\mathcal{R}}_k)\}$ is referred to as the "hierarchical" form of $\{(\mathcal{D}_k, \mathcal{R}_k)\}$.

We refer to [13] for more details, but we do point out that the existence of the limit

function φ_i^k in (101) implies on one hand

$$\mathcal{D}_k \hat{\mathcal{R}}_k = I$$

and also that

$$(\hat{\mathcal{R}}_k \mathcal{D}_k)\hat{\mathcal{R}}_{k-1} = \hat{\mathcal{R}}_{k-1}$$

$$\hat{\mathcal{R}}_k(\mathcal{D}_k f) - \hat{\mathcal{R}}_{k-1}(\mathcal{D}_{k-1} f) = \sum_{i=1}^{J_{k-1}} d_i^k(f)\psi_i^k$$

thus the multi-scale decomposition of $f(x)$

$$\hat{\mathcal{R}}_L(x, \bar{f}^k) = \hat{\mathcal{R}}_0(x, \bar{f}^0) + \sum_{k=1}^{L} Q_k(x, f)$$

where

$$Q_k(x, f) = \hat{\mathcal{R}}_k(x, \bar{f}^k) - \hat{\mathcal{R}}_{k-1}(x, \bar{f}^{k-1})$$

admits a representation in terms of the functions $\{\psi_i^k\}_{i=1}^{J_{k-1}}$. Hence the set $\{\{\psi_i^k\}_{i=1}^{J_{k-1}}\}_{k=1}^{L}$ is referred to as the basis of "generalized wavelets" corresponding to $\{(\mathcal{D}_k, \hat{\mathcal{R}}_k)\}$.

If the sequence $\{(\mathcal{D}, \hat{\mathcal{R}}_k)\}_{k=0}^{+\infty}$ is *complete* in $\mathcal{F}$, i.e., if for all $f \in \mathcal{F}$

$$\lim_{L \to +\infty} ||\hat{\mathcal{R}}_L \mathcal{D}_L f - f|| = 0$$

then

$$\mathcal{B}_M = (\{\{\bar{\psi}_i^k\}_{i=1}^{J_{k-1}}\}_{k=1}^{+\infty}, \{\bar{\varphi}_i^0\}_{i=1}^{J_0})$$

is a basis in $\mathcal{F}$. The expression of any $f \in \mathcal{F}$ in this basis is

$$f = \sum_{m=1}^{+\infty} \sum_{i=1}^{J_{m-1}} d_i^m(f)\psi_i^m + \sum_{i=0}^{J_0} (\mathcal{D}^0 f)_i \varphi_i^0. \tag{105}$$

i.e., the coordinates are the coefficients of the multi-resolution representation of f in the *original* sequence (see [13]).

Notice that $\hat{\mathcal{R}}_k$ is, in general, different from $\mathcal{R}_k$; however, if

$$(\mathcal{R}_k \mathcal{D}_k)\mathcal{R}_{k-1} = \mathcal{R}_{k-1}$$

(this property characterizes "hierachical" reconstructions) then, it is shown in [11, 13] that

$$\hat{\mathcal{R}}_k = \mathcal{R}_k$$

The limiting process is not needed in this case because

$$\varphi_i^{m,L} = \mathcal{R}_L \bar{\varphi}_i^L = \mathcal{R}_m \delta_i^{J_m} = \varphi_i^m, \quad \forall L \geq m.$$

and the multi-resolution representation corresponds to the multi-scale decomposition (102) with $\hat{\mathcal{R}}_k = \mathcal{R}_k$

Let us concentrate now on hat-weighted multi-resolution representations. In the previous section, we defined

$$\mathcal{R}_k(x;\bar{f}^k) = \frac{d^2}{dx^2} I_k(x;H^k) \tag{106}$$

where $I_k(x;H^k)$ is an interpolatory reconstruction of the set $H_j^k = H(x_j^k)$ $1 \le j \le J_k$. Because the reconstruction procedure is based on an interpolation of the point-values of the "second primitive", many properties of the interpolatory multi-resolution setting carry over to this framework.

LEMMA 6.1. *If the interpolatory reconstructions $\hat{I}_k$ are hierarchical, so are the corresponding reconstructions obtained by (106).*

Proof. The hierarchical property of the interpolatory reconstructions means that if

$$\hat{H}_j^k = \hat{I}_{k-1}(x_j^k;H^{k-1}) \qquad 1 \le j \le J_k \tag{107}$$

then

$$\hat{I}_k(x;\hat{H}^k) = \hat{I}_{k-1}(x,H^{k-1}). \tag{108}$$

To prove that

$$\hat{\mathcal{R}}_k(x;\bar{f}^k) = \frac{d^2}{dx^2}\hat{I}_k(x;H^k)$$

is also hierarchical, observe that

$$(\mathcal{D}_k\hat{\mathcal{R}}_{k-1}\bar{f}^{k-1})_j = \frac{1}{h_k^2}\left\{\hat{I}_{k-1}(x_{j-1}^k;H^{k-1}) - 2\hat{I}_{k-1}(x_j^k;H^{k-1}) + \hat{I}_{k-1}(x_{j+1}^k;H^{k-1})\right\}.$$

Thus, the set $\hat{H}_j^k$ in (107) are the values of the "second primitive" of the function $(\hat{\mathcal{R}}_{k-1}\bar{f}^{k-1})(x)$ on the $k-th$ grid. Hence

$$\hat{\mathcal{R}}_k\mathcal{D}_k\hat{\mathcal{R}}_{k-1}\bar{f}^{k-1}(x) = \frac{d^2}{dx^2}\hat{I}_k(x;\hat{H}^k) = \frac{d^2}{dx^2}\hat{I}_{k-1}(x;H^{k-1}) = \hat{\mathcal{R}}_{k-1}\bar{f}^{k-1}(x)$$

which completes the proof. □

The relation between the hat-averaged multi-resolution representations of a function and the associated interpolatory multi-resolution of its "second primitive" can be further exploited. We can obtain multi-scale decompositions of the first from the multi-scale decompositions of the latter.

Assume that

$$\hat{I}_k(x;H^k) = \sum_{j=1}^{J_k} H_j^k\hat{\varphi}_j^k(x); \quad \hat{\varphi}_j^k(x) = \hat{I}_k(x;\delta_j^k) \tag{109}$$

is hierarchical. Then (see [13])

$$\hat{I}_k(x;H^k) - \hat{I}_{k-1}(x;H^{k-1}) = \sum_{j=1}^{J_k}\hat{d}_j^k(H)\hat{\varphi}_{2j-1}^k(x) \tag{110}$$

Differentiating (110) and taking into account that

$$\hat{d}_j^k(f) = -\frac{2}{h_k^2}\hat{d}_j^k(H), \quad 1 \le j \le J_{k-1}$$

we obtain

$$(\hat{\mathcal{R}}_k \bar{f}^k)(x) - (\hat{\mathcal{R}}_{k-1}\bar{f}^{k-1})(x) = -\frac{h_k^2}{2}\sum_{j=1}^{J_{k-1}} \hat{d}_j^k(f)\frac{d^2}{dx^2}\hat{\varphi}_{2j-1}^k(x). \tag{111}$$

Hence

$$(\hat{\mathcal{R}}_k \bar{f}^k)(x) - (\hat{\mathcal{R}}_{k-1}\bar{f}^{k-1})(x) = \sum_{j=1}^{J_k} \hat{d}_j^k(f)\psi_j^k(x), \tag{112}$$

i.e.

$$(\hat{\mathcal{R}}_L \bar{f}^L)(x) = (\hat{\mathcal{R}}_0 \bar{f}^0)(x) + \sum_{k=1}^{L}\sum_{j=1}^{J_k} \hat{d}_j^k(f)\psi_j^k(x). \tag{113}$$

with

$$\psi_j^k(x) = -\frac{h_k^2}{2}\frac{d^2}{dx^2}\hat{\varphi}_{2j-1}^k(x), \quad 1 \le j \le J_{k-1}.$$

Moreover, differentiating (109) we get

$$\begin{aligned}
\hat{\mathcal{R}}_k \bar{f}^k(x) &= \sum_{j=1}^{J_k} H_j^k \frac{d^2}{dx^2}\hat{\varphi}_j^k(x); \\
&= H_1^k \frac{d^2}{dx^2}\hat{\varphi}_1^k(x) + \sum_{j=2}^{J_k}\left(H_1 + \sum_{l=1}^{j-1}(H_{l+1}^k - H_l^k)\right)\frac{d^2}{dx^2}\hat{\varphi}_j^k(x) \\
&= H_1^k \sum_{j=1}^{J_k} j\frac{d^2}{dx^2}\hat{\varphi}_j^k(x) + \sum_{j=1}^{J_k-1}\bar{f}_j^k\left(\sum_{i=j+1}^{J_k} h_k^2\frac{d^2}{dx^2}\hat{\varphi}_i^k(x)\right) \\
&= \sum_{i=1}^{J_k-1}\bar{f}_i^k\varphi_i^k(x)
\end{aligned}$$

where

$$\varphi_i^k(x) = \sum_{l=i+1}^{J_k} h_k^2\frac{d^2}{dx^2}\hat{\varphi}_l^k(x).$$

Note that

$$\sum_{j=1}^{J_k} j\frac{d^2}{dx^2}\hat{\varphi}_j^k(x) = \frac{d^2}{dx^2}I_k(x; \sum_{j=1}^{J_k} j\delta_j^{J_k}) = 0$$

if I_k is at leat first order accurate.

6.1. Piecewise polynomial interpolation. Let $\mathcal{S}$ denote the stencil

$$\mathcal{S} = \mathcal{S}(r,s) = \{-s, -s+1 \cdots, -s+r\} \qquad r \le s \le 0, \quad r \le 1$$

and let $\{L_m(y)\}_{m\in\mathcal{S}}$ denote the Lagrange interpolation polynomials for this stencil

$$L_m(y) = \prod_{j=-s, j\neq m}^{-s+r} \left(\frac{y-j}{m-j}\right), \qquad L_m(i) = \delta_{i,m}, \quad i \in \mathcal{S}.$$

It is clear that

$$q_j^k(x; H^k, r, s) = \sum_{m=-s}^{-s+r} H_{j+m}^k L_m\left(\frac{x - x_j^k}{h_k}\right)$$

interpolates $H(x)$ at the points $\{x_{j-s}^k, \cdots, x_{j-s+r}^k\}$. Thus,

$$I_k(x, H^k) = q_j^k(x; H^k, r, s) \qquad x \in [x_{j-1}^k, x_j^k], \quad 1 \le j \le J_k + 1$$

is a piecewise polynomial function that interpolates $H(x)$ at the points $\{x_j^k\}_{j=1}^{J_k+1}$. Using the reconstruction via "second primitive" technique, described in section 5, the reconstruction operator is defined as

$$(\mathcal{R}_k \bar{f}^k)(x) = \tilde{I}_k(x; H^k) + \sum_{j=0}^{J_k} \delta(x - x_j^k) s_j^k$$

where

$$\tilde{I}_k(x; H^k) = \frac{d^2}{dx^2} q_j^k(x; H^k, r, s) \qquad x \in [x_{j-1}^k, x_j^k]$$

and

$$s_j^k = \left[\frac{d}{dx} q_{j+1}^k(x; H^k, r, s) - \frac{d}{dx} q_j^k(x; H^k, r, s)\right]_{x=x_j^k}.$$

The prediction operators are given by (82). It should be noted that when I_k is obtained by piecewise polynomial interpolation, the prediction operators can always be expressed in terms of the $\bar{f}^k$ without explicitly using (and computing) the values H^k.

Here, we shall consider only centered interpolation procedures. They correspond to situations where the interpolatory stencil is symmetric around the given interval.

If $r = 2s + 2$, $q_j^k(x; H^k, r, s)$ is the unique polynomial of degree $2s + 1$ that interpolates $H(x)$ at the $2s + 2$ grid points $\{x_{j-s-1}^k, \cdots, x_{j+s}^k\}$. Notice that, formally,

$$I_k(x, H^k) = H(x) + O(h_k)^{2s+2}$$

thus, differentiating twice (in regions of smoothness)

$$\mathcal{R}_k(x; \bar{f}^k) = f(x) + O(h_k)^{2s}.$$

Hence, the reconstruction thus obtained is, formally, of order $2s$. In what follows, we denote by p the order of (formal) accuracy of $\mathcal{R}_k$.

Following steps similar to those in [12, 13] we get the following encoding and decoding

algorithms for reconstructions of order p, derived from centered interpolation procedures of order $p+2$.

$\mu(\bar{f}^L) = M\bar{f}^L$ (Encoding)

$$(114)\qquad \begin{cases} \text{Do} \quad k = L, 1 \\ \qquad \bar{f}_i^{k-1} = \frac{1}{4}(\bar{f}_{2i-1}^k + 2\bar{f}_{2i}^k + \bar{f}_{2i+1}^k), & 1 \le i \le J_{k-1} \\ \qquad d_i^k = \bar{f}_{2i-1}^k - \sum_{l=1}^s \beta_l(\bar{f}_{i+l-1}^{k-1} + \bar{f}_{i-l}^{k-1}), & 1 \le i \le J_{k-1} \end{cases}$$

$\bar{f}^L = M^{-1}\mu(\bar{f}^L)$ (Decoding)

$$(115)\qquad \begin{cases} \text{Do} \quad k = 1, L \\ \qquad \bar{f}_{2i-1}^k = d_i^k + \sum_{l=1}^s \beta_l(\bar{f}_{i+l-1}^{k-1} + \bar{f}_{i-l}^{k-1}), & 1 \le i \le J_{k-1} \\ \qquad \bar{f}_{2i}^k = 2\bar{f}_i^{k-1} - \frac{1}{2}(\bar{f}_{2i-1}^k + \bar{f}_{2i+1}^k), & 1 \le i \le J_{k-1} \end{cases}$$

where

$$(116)\qquad \begin{cases} p = 2 \;\Rightarrow\; \beta_1 = \frac{1}{2} \\ p = 4 \;\Rightarrow\; \beta_1 = \frac{19}{32}, \beta_2 = -\frac{3}{32} \\ p = 6 \;\Rightarrow\; \beta_1 = \frac{162}{256}, \beta_2 = -\frac{39}{256}, \beta_3 = \frac{5}{256} \end{cases}$$

We are considering only the periodic case, thus $\bar{f}_{J_k+i}^k = \bar{f}_i^k \ \forall k, i$. In the non periodic case one possibility is to lower the order of the interpolation procedure near the boundaries; another possibility is use extrapolation in order to get the extra values needed for the interpolatory reconstruction.

Let us consider now the limiting process $L \to \infty$ in (101)

$$\varphi_i^{m,L}(x) = \mathcal{R}_L(x; \bar{\varphi}_i^{m,L}) = \mathcal{R}_L A_m^L \delta_i^m$$

where we start by setting

$$\bar{\varphi}_i^{m,m} = \delta_i^{J_m}$$

at the points of the m-th grid and then repeatedly apply P_k^{k+1}.

It is proven in [13] that, when the linear operators $\mathcal{R}_k$ are translation invariant, i.e.

$$(117)\qquad (\mathcal{R}_k \delta_{j-q}^{J_k})(x - qh_k) = (\mathcal{R}_k \delta_j^{J_k})(x) \qquad \forall q$$

and the same for all levels of resolution, i.e.

$$(118)\qquad (R_{k+1}\delta_0^{J_{k+1}})(x) = (\mathcal{R}_k \delta_0^{J_k})(2x) \qquad \forall q$$

convergence of the single sequence

$$(119)\qquad \mathcal{R}_L A_0^L \delta_0^{J_0} \longrightarrow \varphi_0$$

guarantees that

$$\varphi_i^{m,L} \longrightarrow \varphi_i^m.$$

Moreover

$$\varphi_i^m = \varphi(\frac{x}{h_m} - i)$$

where

$$\varphi(x) = \varphi_0(xh_0) \tag{120}$$

In the periodic case, we take the same stencil $\forall j, k$ thus, convergence of the single sequence

$$\mathcal{R}_L \bar{\varphi}_0^{0,L} = \mathcal{R}_L A_0^L \delta_0^{J_0} \longrightarrow \varphi_0 \tag{121}$$

implies the existence of the hierarchical reconstructions $\hat{\mathcal{R}}_k$.

The continuous differentiability of the limit functions in the interpolatory case, can be used in the cell-average context to prove uniform convergence of (121) to a continuous function (see [12], Appendix B)

If the limit function is at least C^2, a similar argument proves uniform convergence of (121) to a continuous function in the case of hat-weighted averages. We sketch a proof of this fact in Appendix B.

6.2. Cubic Splines. Let $I_k(x; H^k)$ be the unique piecewise-cubic function (i.e. I_k is a cubic polynomial in each $[x_{j-1}^k, x_j^k]$, $1 \le j \le J_k$) which satisfies

1. $I_k(x_j^k; H^k) = H_j^k$, $1 \le j \le J_k$
2. $\frac{d^l}{dx^l} I_k(x_j^k - 0; H^k) = \frac{d^l}{dx^l} I_k(x_j^k + 0; H^k)$ $l = 1, 2$, $1 \le j \le J_k - 1$
3. $\frac{d^l}{dx^l} I_k(0; H^k) = \frac{d^l}{dx^l} I_k(1; H^k)$, $\quad l = 1, 2.$

In each interval $[x_{j-1}^k, x_j^k]$, $1 \le j \le J_k$, $I_k(x; H^k)$ has the form

$$I_k(x; H^k) = M_{j-1} \frac{(x_j^k - x)^3}{6h_k} + M_j \frac{(x - x_{j-1}^k)^3}{6h_k} + A_j(x - x_{j-1}^k) + B_j \tag{122}$$

where

$$B_j = H_{j-1}^k - M_{j-1} \frac{h_k^2}{6} \qquad A_j = \frac{H_j^k - H_{j-1}^k}{h_k} - \frac{h_k}{6}(M_j - M_{j-1})$$

and $M_1, M_2, \ldots, M_{J_k} (= M_0)$ solve the system:

$$(123)\quad \begin{bmatrix} 2 & \frac{1}{2} & 0 & \dots & \dots & 0 & \frac{1}{2} \\ \frac{1}{2} & 2 & \frac{1}{2} & 0 & \dots & \dots & 0 \\ 0 & \frac{1}{2} & \ddots & \ddots & 0 & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & \\ & & & & & \ddots & 0 \\ 0 & & & & \ddots & \ddots & \frac{1}{2} \\ \frac{1}{2} & 0 & \dots & & 0 & \frac{1}{2} & 2 \end{bmatrix} \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ \\ \\ \vdots \\ M_{J_k} \end{bmatrix} = 3 \begin{bmatrix} \frac{H_0^k - 2H_1^k + H_2^k}{h_k^2} \\ \frac{H_1^k - 2H_2^k + H_3^k}{h_k^2} \\ \vdots \\ \\ \\ \vdots \\ \frac{H_{J_k-1}^k - 2H_{J_k}^k + H_1^k}{h_k^2} \end{bmatrix} = 3 \begin{bmatrix} \bar{f}_1^k \\ \bar{f}_2^k \\ \vdots \\ \\ \\ \vdots \\ \bar{f}_{J_k}^k \end{bmatrix}$$

Since the reconstruction is the second derivate of the interpolatory cubic spline, we only need the coefficients M_j which can be computed in $O(J_k)$ operations using the LU decomposition of the coefficient matrix (see Appendix A).

The cubic-spline interpolation is hierarchical (see [13]), hence $\hat{I}_k = I_k$, $\hat{\mathcal{R}}_k = \mathcal{R}_k$ and the limiting process is not needed. Moreover, it is translation invariant in the periodic case, therefore $\hat{\varphi}_j^k(x)$ can be described in terms of the single function $\hat{\varphi}_0^k(x) = I_k(x; \delta_0^k)$, for each k, by

$$\hat{\varphi}_j^k(x) = \hat{\varphi}_0^k(x - ih_k).$$

The corresponding 'hat' multi-scale decomposition of f can then be easily obtained.

7. Data Compression. Multi-resolution representations of a set of data enable us to obtain data compression by replacing $\mu(\bar{f}^L)$ with a truncated $\hat{\mu}(\bar{f}^L)$

$$(124)\quad \hat{\mu}(\bar{f}^L) = \mathbf{tr}_\epsilon \cdot \mu(\bar{f}^L) = \begin{pmatrix} \hat{d}^L \\ \vdots \\ \hat{d}^1 \\ \bar{f}^0 \end{pmatrix} \in V^L.$$

where

$$(125)\quad (\hat{d}^k)_j = \mathbf{tr}(d_j^k; \epsilon_k) = \begin{cases} 0 & |d_j^k| \le \epsilon_k \\ d_j^k & \text{otherwise} \end{cases}$$

The crucial numerical issue is the stability of the data-compression procedure. We would like to formulate conditions on $\mathcal{D}_k$, $\mathcal{R}_k$ and ϵ_k so that

$$(126)\quad \|\bar{f}^L - \hat{f}^L\| \le C \cdot \epsilon, \qquad \bar{f}^L = M^{-1} \cdot \mathbf{tr}_\epsilon(M \bar{f}^L),$$

where C is independent of L.

7.1. Linear Reconstruction operators. Let $\hat{e}^k$ denote

$$(127)\quad \hat{e}^k = T_{k-1}^k(d^k - \hat{d}^k),$$

then

$$(128)\quad \bar{f}^L - \hat{f}^L = \sum_{m=1}^{L} A_m^L \hat{e}^m;$$

this shows that

$$||A_m^L \hat{e}^m|| \leq C\epsilon_m, \qquad \sum_{m=1}^{L} \epsilon_m \leq \epsilon, \tag{129}$$

is a sufficient condition for (126).

By Lax's equivalence theorem, convergence is equivalent to stability + consistency. Therefore, convergence of the limiting process

$$\mathcal{R}_L A_m^L \delta_i^m \longrightarrow \varphi_i^m$$

also implies stability of the data-compression procedure.

7.2. Error Control. The strategy (124) gives us direct control over the rate of compression through an appropriate choice of the tolerance levels $\{\epsilon_k\}_{k=0}^{L}$. However once we use the truncated values (125) in the corresponding decoding algorithm or multi-resolution representation of $f(x)$ we get an error which can be estimated by analysis but cannot be directly controlled. This strategy is therefore suitable for applications where we are limited in capacity and we have to settle for whatever quality is possible under this limitation.

There are other applications where quality control is of utmost importance, yet we would like to be as economical as possible with respect to storage and speed of computation. To accomplish this goal we present a modification of the encoding algorithm which keeps track of the cumulative error in a predetermined decoding procedure and truncates accordingly. This enables us to specify the desired level of accuracy in the decomposed signal as well as in the reduced functional representation. As to be expected (from considerations of the uncertainty principle), we cannot specify compression rate at the same time.

Given any tolerance level ϵ for accuracy, our task is to come up with a compressed representation

$$\{(\tilde{d}^L, \ldots, \tilde{d}^1), f^0\} \tag{130}$$

such that

$$\| f^L - \tilde{f}^L \|_\infty = \max_{1 \leq i \leq J_L} |f_i^L - \tilde{f}_i^L| \leq C\epsilon \tag{131}$$

for $\tilde{f}^L$ which is obtained by decoding the compressed multi-resolution representation. A modified encoding procedure that accomplishes this task is described algorithmically as follows:

1. Compute the multi-resolution analysis of the input data by

$$\begin{cases} DO \quad k = L, 1 \\ \quad \begin{cases} DO \quad j = 1, J_k \\ \quad \bar{f}_j^{k-1} = \frac{1}{4}(\bar{f}_{2j-1}^k + 2\bar{f}_{2j}^k + \bar{f}_{2j+1}^k) \end{cases} \end{cases} \tag{132}$$

2. Set

$$\tilde{f}^0 = \bar{f}^0 \tag{133}$$

3. Calculate

$$
\left\{
\begin{array}{l}
DO \quad k = 1, L \\
\bar{f}^{PR}_{J_{k-1}+1} = (\mathcal{D}_k \mathcal{R}_{k-1} \tilde{f}^{k-1})_{2J_{k-1}+1} \\
\tilde{d}^k_{J_{k-1}+1} = tr(\bar{f}^k_{2J_{k-1}+1} - \bar{f}^{PR}_{J_{k-1}+1} - (\bar{f}^{k-1}_{J_{k-1}} - \tilde{f}^{k-1}_{J_{k-1}}), \epsilon_k) \\
\tilde{f}^k_{2J_{k-1}+1} = \bar{f}^{PR}_{J_{k-1}+1} + \tilde{d}^k_{J_{k-1}+1} \\
\left\{
\begin{array}{l}
DO \quad j = J_{k-1}, 1, -1 \\
\bar{f}^{PR}_j = (\mathcal{D}_k \mathcal{R}_{k-1} \tilde{f}^{k-1})_{2j-1} \\
\tilde{d}^k_j = tr((\bar{f}^k_{2j-1} - \bar{f}^{PR}_j) - (\bar{f}^{k-1}_j - \tilde{f}^{k-1}_j), \epsilon_k) \\
\tilde{f}^k_{2j-1} = \bar{f}^{PR}_j + \tilde{d}^{k-1}_j \\
\tilde{f}^k_{2j} = 2\tilde{f}^{k-1}_j - \frac{1}{2}(\tilde{f}^k_{2j-1} + \tilde{f}^k_{2j+1})
\end{array}
\right.
\end{array}
\right. \tag{134}
$$

Let us denote the error at the k-th level by $\bar{E}^k_j$, i.e.

$$\bar{E}^k_j = \bar{f}^k_j - \tilde{f}^k_j \tag{135}$$

and the prediction error by $\bar{E}^{PR}_j$, i.e.

$$\bar{E}^{PR}_j = \bar{f}^k_{2j-1} - \bar{f}^{PR}_j \tag{136}$$

Let us prove that

$$|\bar{E}^k_{2j-1}|_\infty \leq \epsilon_k + \| \bar{E}^{k-1} \|_\infty, \quad j = J_{k-1} + 1, 1, -1 \tag{137}$$

In fact, since

$$\bar{E}^k_{2J_{k-1}+1} = \bar{E}^{PR}_{J_{k-1}+1} - tr(\bar{E}^{PR}_{J_{k-1}+1} - \bar{E}^{k-1}_{J_{k-1}}, \epsilon_k) \tag{138}$$

we have

$|\bar{E}^{PR}_{J_{k-1}+1} - \bar{E}^{k-1}_{J_{k-1}}| > \epsilon_k \Rightarrow |\bar{E}^k_{2J_{k-1}+1}| = |\bar{E}^{k-1}_{J_{k-1}}| \leq \| \bar{E}^{k-1} \|_\infty,$

$|\bar{E}^{PR}_{J_{k-1}+1} - \bar{E}^{k-1}_{J_{k-1}}| \leq \epsilon_k \Rightarrow$

$$|\bar{E}^k_{2J_{k-1}+1}| = |\bar{E}^{PR}_{J_{k-1}+1}| \leq |\bar{E}^{k-1}_{J_{k-1}}| + |\bar{E}^{PR}_{J_{k-1}+1} - \bar{E}^{k-1}_{J_{k-1}}| \leq \| \bar{E}^{k-1} \|_\infty + \epsilon_k.$$

On the other hand,

$$\bar{E}^k_{2j-1} = \bar{E}^{PR}_j - tr(\bar{E}^{PR}_j - \bar{E}^{k-1}_j, \epsilon_k) \tag{139}$$

for each $j = J_{k-1}, 1, -1$, thus

$|\bar{E}^{PR}_j - \bar{E}^{k-1}_j| > \epsilon_k \Rightarrow |\bar{E}^k_{2j-1}| = |\bar{E}^{k-1}_j| \leq \| \bar{E}^{k-1} \|_\infty$

and

$|\bar{E}^{PR}_j - \bar{E}^{k-1}_j| \leq \epsilon_k, \Rightarrow$

$$|\bar{E}^k_{2j-1}| = |\bar{E}^{PR}_j| \leq |\bar{E}^{PR}_j - \bar{E}^{k-1}_j| + |\bar{E}^{k-1}_j| \leq \epsilon_k + \| \bar{E}^{k-1} \|_\infty .$$

Let us prove now that

$$|\bar{E}^k_{2j}| \leq \epsilon_k + 2 \parallel \bar{E}^{k-1} \parallel_\infty \quad j = 1, J_{k-1} \tag{140}$$

Since

$$\bar{E}^k_{2j} = \bar{E}^{k-1}_j - \frac{1}{2}(\bar{E}^k_{2j-1} + \bar{E}^k_{2j+1}) \tag{141}$$

$|\bar{E}^{PR}_j - \bar{E}^{k-1}_j| \leq \epsilon_k \Rightarrow$

$$\begin{aligned} |\bar{E}^k_{2j}| &= |2\bar{E}^{k-1}_j - \frac{1}{2}(\bar{E}^{PR}_j + \bar{E}^k_{2j+1})| \\ &= |-\frac{1}{2}(\bar{E}^{PR}_j - \bar{E}^{k-1}_j) + \frac{3}{2}\bar{E}^{k-1}_j - \frac{1}{2}\bar{E}^k_{2j+1}| \\ &\leq \frac{1}{2}\epsilon_k + \frac{3}{2} \parallel \bar{E}^{k-1} \parallel_\infty + \frac{1}{2}(\epsilon_k + \parallel \bar{E}^{k-1} \parallel_\infty) \\ &\leq \epsilon_k + 2 \parallel \bar{E}^{k-1} \parallel_\infty, \end{aligned}$$

and
$|\bar{E}^{PR}_j - \bar{E}^{k-1}_j| > \epsilon_k \Rightarrow$

$$\begin{aligned} |\bar{E}^k_{2j}| &= |2\bar{E}^{k-1}_j - \frac{1}{2}(\bar{E}^{k-1}_j + \bar{E}^k_{2j+1})| \leq \frac{3}{2}|\bar{E}^{k-1}_j| + \frac{1}{2}|\bar{E}^k_{2j+1}| \\ &\leq 2 \parallel \bar{E}^{k-1} \parallel_\infty + \frac{1}{2}\epsilon_k. \end{aligned}$$

Recalling that $\bar{E}^0 = 0$, we get from (137) and (140)

$$\parallel \bar{E}^L \parallel_\infty \leq \epsilon_L + 2 \parallel \bar{E}^{L-1} \parallel_\infty \leq \ldots \leq \sum_{l=1}^{L} 2^{L-l}\epsilon_l; \tag{142}$$

taking

$$\epsilon_l = \epsilon / 2^{L-l+1} \tag{143}$$

we get

$$\parallel \bar{f}^L - \tilde{f}^L \parallel_\infty = \parallel \bar{E}^L \parallel_\infty \leq \frac{L}{2}\epsilon \tag{144}$$

If the reconstruction operators $\mathcal{R}_k$ are linear functionals, the error-control technique we have described allows us to control the quality of the decoded data instead of the compression rate. If the reconstruction operators are non linear (data dependent) this algorithm guarantees the stability of the data compression procedure, while a simpler algorithm like (43),(44) does not.

From the above analysis it seems that a sensible choice for the tolerance levels is

$$\epsilon_k = \epsilon_{k+1}/2 \tag{145}$$

In our numerical experiments we observe that the computed error bound is much closer to ϵ than to $\epsilon L/2$.

7.3. Non Linear Reconstruction Operators. It is clear that the accuracy of the reconstruction technique plays a key role in the efficiency of data compression algorithms of the type (124). The scale coefficients are directly related to the prediction errors, which measure our success in using the reconstruction procedure to climb up the ladder from low-resolution to high-resolution levels.

For piecewise smooth signals with a finite number of singularities, adaptive, data dependent reconstructions of order p manage to keep the relation

$$\mathcal{R}_k \bar{f}^k(x) = f(x) + O(h_k^p)$$

valid over a larger region than linear (= data independent) reconstructions of the same order.

For discontinuous piecewise smooth signals, data compression algorithms based on cell-averaged multi-resolution and ENO reconstructions give much better compression rates than the corresponding algorithms with linear reconstructions (see [11, 7]). Moreover, the Subcell-Reslution technique of [10] applied to the ENO reconstructions gives maximal compression of piecewise polynomial functions with a finite number of discontinuities.

In the same fashion, ENO reconstruction techniques in the hat-averaged multi-resolution context lead to more efficient data compression algorithms for piecewise smooth signals with a finite number of δ-singularities. The space of such functions is used in vortex methods for the numerical solution of fluid dynamics problems.

It is conceptually easy to modify an ENO reconstruction to account for singularities within a cell. This is the so-called Subcell Resolution technique ([10, 11]). Let us describe a SR technique in the context of hat-weighted multi-resolution.

Assume

$$f(x) = P(x) + \delta(x - x_d)$$

where $P(x)$ is a smooth function and $x_d \in (x_{j-1}^k, x_j^k)$. We know that $\mathcal{R}_k \bar{f}^k$ is given by (79),(80), thus it is represented by a smooth function in the interval $[x_{j-1}^k, x_j^k]$ and hence, it cannot accurately represent $f(x)$ in this interval. The second primitive of $f(x)$ will have a discontinuous derivative at x_d, thus if there is enough resolution on the k-th grid, we expect $I_{k,L}(x) = I_{k,j-1}(x; H^k)$ and $I_{k,R}(x) = I_{k,j+1}(x; H^k)$ to intersect at a point, $\tilde{x}_d$, in $[x_{j-1}^k, x_j^k]$. Moreover

$$\tilde{x}_d - x_d = O(h^{p+2}).$$

where p is the order of the reconstruction.

Thus, if we replace (77) by

$$I_k^{SR}(x; H^k) = \begin{cases} I_k(x; H^k) & \text{for } x \notin [x_{j-1}^k, x_j^k] \\ I_{k,L}(x) & \text{for } x \in [x_{j-1}^k, \tilde{x}_d] \\ I_{k,R}(x) & \text{for } x \in [\tilde{x}_d, x_j^k] \end{cases} \tag{146}$$

and (79) by

$$\tilde{\mathcal{R}}_k \bar{f}^k(x) = \tilde{\tilde{I}}(x) + \sum_{\substack{l=1 \\ l \neq j-1, j}}^{J_k} s_l^k \delta(x - x_l^k) + \tilde{s}_j^k \delta(x - \tilde{x}_d) \tag{147}$$

with

$$\tilde{s}_j^k = \left[\frac{d}{dx} I_{k,R}(x) - \frac{d}{dx} I_{k,L}(x)\right]_{x=\tilde{x}_d},$$

and

$$\tilde{\tilde{I}}_k^{SR}(x; H^k) = \begin{cases} \frac{d^2}{dx^2} I_k(x; H^k) & \text{for } x \notin [x_{j-1}^k, x_j^k] \\ \frac{d^2}{dx^2} I_{k,L}(x) & \text{for } x \in [x_{j-1}^k, \tilde{x}_d] \\ \frac{d^2}{dx^2} I_{k,R}(x) & \text{for } x \in [\tilde{x}_d, x_j^k] \end{cases} \tag{148}$$

we get a subcell resolution technique which is exact when $P(x)$ is a polynomial function of lesser or equal degree than the reconstructing procedure.

We remark again that, for non linear reconstruction procedures, in order to ensure the numerical stability of the data compression algorithm, one must use an encoding algorithm that implements an error-control policy of the type described in the last paragraph,

8. Numerical Experiments. In this section, we carry out several experiments to test the performance of the different compression algorithms, as well as to compare their performances

In all our experiments the finest grid X^L is a uniform grid of $J_L = 1024$ points.

In the figures

- **tol** is the tolerance on the finest level, ϵ_L. The scale coefficients are truncated according to the following choice of tolerance levels:

point-values	$\epsilon_k = \epsilon_{k+1}$	$\forall k$
cell-averages	$\epsilon_k = \epsilon_{k+1}/2$	$\forall k$
hat-averages	$\epsilon_k = \epsilon_{k+1}/2$	$\forall k$

- **nz** is the number of non-zero elements in the compressed representation.
- **l1** is the L_1 norm of the difference (126).
- **sup** is the L_∞ norm of the difference (126).

The figures show the reconstructed signal and the error. The error is shifted down for displaying purposes. We also display the position of the scale coefficients that are above the specified tolerances in the truncation operation, at each level of resolution. It is interesting to observe the refinement pattern of the scale coefficient for various values of ϵ_L.

8.1. Linear reconstruction Operators. We compare the performance of the algorithms derived from symmetric centered piecewise polynomial interpolation. The degree of the reconstruction procedure is $r = 4$ for the hat and interpolatory multi-resolution and $r = 3$ for the cell-average multi-resolution.

8.1.1. Smooth Functions. We consider the functions

$$f_1(x) = \sin(2\pi x) \tag{149}$$

$$f_2(x) = e^{-300(x-.5)^2} \tag{150}$$

The discrete set to which we apply the compression algorithm is $\{f(x_j^L)\}_{j=1}^{J_L}$.

Tables 1 and 2 compare the performances of the three compression algorithms for different tolerances. For smooth functions, the three algorithms behave approximately alike. The algorithm based on point values, as expected, gives the best relation between the quality of

the recovered signal and the compression rate obtained.

8.1.2. Discontinuous, Piecewise Smooth Functions. Consider the functions

$$f_3(x) = \begin{cases} 4x & 0 \le x \le 1/4 \\ 2 - 4x & 1/4 < x \le 1/2 \\ -1 & 1/2 < x \le 3/4 \\ 4x - 4 & 3/4 < x \le 1 \end{cases} \tag{151}$$

$$f_4(x) = \begin{cases} \sin(\pi x) & 0 \le x \le 1/2 \\ -\sin(\pi x) & 1/2 < x \le 1 \end{cases} \tag{152}$$

Table 3 and 4 compare the performances of the three compression algorithms for different tolerances. The cell-averaged algorithm gives now (as expected) the best relation between quality and compression rate. The hat-averaged algorithm seems to be the less efficient for $f_3(x)$ than for $f_4(x)$.

8.1.3. Piecewise Smooth Distributions. Next, we carry out several experiments with distributions of the type

$$g(x) = f_i(x) + \alpha\delta(x - \beta) + \alpha\delta(x - \gamma) \tag{153}$$

In our experiments we have taken $\alpha = 1.9 \cdot 10^{-4}$, $\beta = .125$ and $\gamma = .5$ or $\gamma = .625$. $f_i(x)$ $i = 1, \cdots, 4$ are the functions described before.

Tables 5 to 10 compare the performances of the compression algorithms based on cell-averages and hat-averages, for different tolerances. Observe that point-values of the function lack now meaning. For $f_1(x)$ and $f_2(x)$, both algorithms are comparable, with a slight advantage for the hat-averaged one. For $f_3(x)$ and $f_4(x)$, and probably due to the discontinuous nature of the function, the cell-averaged algorithm comes slightly ahead again. The behavior observed in tables 9 and 10 seems to indicate that it is more efficient to compress with cell-averages when dealing with discontinuous functions.

8.2. Non Linear Reconstruction Operators. When we use a data-dependent interpolatory technique, the reconstruction operators thus obtained are non-linear. As we have mentioned before, we then need an error-controlled compression mechanism to guarantee numerical stability. In tables 11 to 13, we compile the results of two sets of experiments, one with the error-controlled algorithm of section 7.2 (labeled $EC2$) and the other with the original algorithm (114),(115). Algorithm $EC2$ always keeps the error below ϵ_L, even though the theoretical bound we found is larger. When the error in the unmodified algorithm is below ϵ_L, the error-control mechanism does not have a significant effect. Again, we observe that the cell-average and the hat-average algorithms give similar results, although the cell-average algorithm is slightly more efficient.

In both cases, we have chosen nonlinear ENO reconstructions with order of accuracy $r = 4$

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
point	.1	3	$4.7730E-02$	$8.2149E-02$
	.05	7	$3.0309E-03$	$7.8261E-03$
	.01	7	$3.0309E-03$	$7.8261E-03$
	.005	11	$1.0763E-03$	$3.2417E-03$
	.001	15	$1.8940E-04$	$5.3969E-04$
cell	$.1/2^k$	25	$5.0856E-04$	$2.1660E-03$
	$.05/2^k$	27	$3.3909E-04$	$8.1390E-04$
	$.01/2^k$	39	$1.4941E-04$	$4.3417E-04$
	$.005/2^k$	55	$4.3022E-05$	$1.4423E-04$
	$.001/2^k$	63	$2.8228E-05$	$6.1482E-05$
hat	$.1/2^k$	15	$8.3538E-04$	$3.1567E-03$
	$.05/2^k$	19	$5.2233E-04$	$2.1846E-03$
	$.01/2^k$	31	$5.3635E-05$	$1.9626E-04$
	$.005/2^k$	31	$5.3635E-05$	$1.9626E-04$
	$.001/2^k$	51	$1.1419E-05$	$7.3257E-05$

TABLE 1
$f_1(x) = \sin(2\pi x)$

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
point	.1	8	$3.0217E-02$	$6.2520E-02$
	.05	12	$6.8698E-03$	$4.5750E-02$
	.01	20	$8.7457E-04$	$9.6547E-03$
	.005	24	$3.0540E-04$	$1.5793E-03$
	.001	30	$1.8165E-04$	$8.75919E-04$
cell	$.1/2^k$	24	$6.7285E-04$	$4.5457E-03$
	$.05/2^k$	32	$2.8498E-04$	$2.4714E-03$
	$.01/2^k$	46	$8.4476E-05$	$6.0450E-04$
	$.005/2^k$	50	$5.9384E-05$	$6.0450E-04$
	$.001/2^k$	68	$2.1614E-05$	$1.6485E-04$
hat	$.1/2^k$	28	$7.1022E-04$	$6.9823E-03$
	$.05/2^k$	30	$4.8223E-04$	$6.4685E-03$
	$.01/2^k$	44	$7.9450E-05$	$8.3089E-04$
	$.005/2^k$	46	$6.6942E-05$	$7.4236E-04$
	$.001/2^k$	60	$2.3336E-05$	$2.6943E-04$

TABLE 2
$f_2(x) = e^{-300(x-.5)^2}$

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
point	.05	27	$1.0159E-02$	$6.4920E-02$
	.01	36	$5.2057E-04$	$8.5144E-03$
	.005	40	$1.2916E-04$	$4.2572E-03$
	.001	48	$6.9141E-06$	$9.7656E-04$
	.0005	52	$9.5367E-07$	$4.8828E-04$
cell	$.1/2^k$	36	$2.0188E-04$	$1.0254E-02$
	$.05/2^k$	38	$1.0014E-04$	$9.8877E-03$
	$.01/2^k$	42	$2.4319E-05$	$4.3945E-03$
	$.005/2^k$	44	$1.1921E-05$	$2.1973E-03$
	$.001/2^k$	48	$1.4305E-06$	$4.8828E-04$
hat	$.1/2^k$	47	$5.2200E-04$	$2.3346E-02$
	$.05/2^k$	51	$2.8175E-04$	$2.3346E-02$
	$.01/2^k$	60	$5.6258E-05$	$2.9765E-03$
	$.005/2^k$	64	$2.6232E-05$	$1.4882E-03$
	$.001/2^k$	72	$5.7146E-06$	$5.5308E-04$

TABLE 3
$f_3(x)$

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
point	.1	27	$1.2392E-02$	$8.0562E-02$
	.05	28	$5.3386E-04$	$3.2432E-03$
	.01	28	$5.9263E-05$	$3.2432E-03$
	.001	29	$9.0928E-05$	$4.5755E-04$
cell	$.1/2^k$	28	$2.2198E-04$	$4.9411E-04$
	$.05/2^k$	28	$2.2198E-04$	$4.9411E-04$
	$.01/2^k$	40	$3.9263E-05$	$1.4447E-04$
	$.001/2^k$	58	$1.0612E-05$	$4.1444E-05$
hat	$.1/2^k$	35	$3.3525E-04$	$4.6874E-02$
	$.05/2^k$	38	$7.3384E-05$	$3.6074E-04$
	$.01/2^k$	40	$3.3500E-05$	$1.6687E-04$
	$.001/2^k$	50	$3.3684E-06$	$1.1659E-05$

TABLE 4
$f_4(x)$, *discontinuous sine*

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
cell	$.1/2^k$	36	$6.999E-04$	$2.4458E-02$
	$.05/2^k$	37	$6.1519E-04$	$2.4458E-02$
	$.03/2^k$	43	$4.7634E-04$	$2.3733E-02$
	$.01/2^k$	67	$1.2239E-04$	$4.3417E-04$
	$.005/2^k$	80	$3.9614E-05$	$1.4423E-04$
hat	$.1/2^k$	37	$1.1733E-03$	$1.0476E-02$
	$.05/2^k$	45	$7.3826E-04$	$2.5341E-02$
	$.03/2^k$	49	$5.0460E-04$	$8.4232E-03$
	$.01/2^k$	72	$4.4683E-05$	$1.9626E-04$
	$.005/2^k$	72	$4.4682E-05$	$1.9626E-04$

TABLE 5
Sine function + delta(x-.125) + delta(x-.5)

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
cell	$.1/2^k$	35	$9.2362E-04$	$2.4585E-02$
	$.05/2^k$	42	$5.7921E-04$	$2.4586E-02$
	$.02/2^k$	64	$1.9417E-04$	$1.2656E-03$
	$.01/2^k$	73	$9.2170E-05$	$6.0450E-04$
	$.001/2^k$	96	$2.0344E-05$	$1.6486E-04$
hat	$.1/2^k$	47	$1.1279E-03$	$2.0708E-02$
	$.05/2^k$	51	$8.1026E-04$	$8.4330E-03$
	$.02/2^k$	60	$5.1580E-04$	$8.4247E-03$
	$.01/2^k$	82	$6.8672E-05$	$8.3087E-04$
	$.001/2^k$	95	$1.7740E-05$	$2.5297E-04$

TABLE 6
Exponential + delta(x-.125) + delta(x-.5)

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
cell	$.1/2^k$	43	$5.5871E-04$	$2.4533E-02$
	$.05/2^k$	45	$4.5697E-04$	$2.4532E-02$
	$.01/2^k$	68	$2.4319E-05$	$4.3945E-03$
	$.001/2^k$	74	$1.4305E-06$	$4.8828E-04$
hat	$.1/2^k$	57	$7.3806E-04$	$1.4190E-02$
	$.05/2^k$	61	$4.9782E-04$	$1.4190E-02$
	$.01/2^k$	81	$6.2399E-05$	$3.1433E-03$
	$.001/2^k$	94	$5.7145E-06$	$5.5308E-04$

TABLE 7
$f_3(x)$ + delta(x-.125) + delta(x-.5)

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
cell	$.1/2^k$	34	$4.6299E-04$	$2.4807E-02$
	$.02/2^k$	52	$1.3724E-04$	$4.3398E-04$
	$.01/2^k$	60	$2.9982E-05$	$1.4447E-04$
	$.001/2^k$	75	$1.0198E-05$	$4.1444E-05$
hat	$.1/2^k$	47	$5.3402E-04$	$3.7719E-02$
	$.02/2^k$	52	$2.5593E-04$	$8.4170E-03$
	$.01/2^k$	63	$3.0284E-05$	$1.6695E-04$
	$.001/2^k$	71	$2.9414E-06$	$1.3103E-05$

TABLE 8
Discontinuous sine + delta(x-.125) + delta(x-.5)

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
cell	$.1/2^k$	40	$4.9675E-04$	$2.4808E-02$
	$.02/2^k$	64	$1.0880E-04$	$4.3398E-04$
	$.01/2^k$	70	$2.8784E-05$	$1.4305E-04$
	$.001/2^k$	85	$9.0541E-05$	$4.1444E-05$
hat	$.1/2^k$	58	$7.3280E-04$	$4.6875E-02$
	$.02/2^k$	63	$4.6797E-04$	$4.4171E-03$
	$.01/2^k$	85	$2.1512E-05$	$1.6491E-04$
	$.001/2^k$	91	$2.3517E-06$	$1.3103E-05$

TABLE 9
Dscontinuous sine + delta(x-.125) + delta(x-.625)

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
cell	$.1/2^k$	49	$5.9388E-04$	$2.4532E-02$
	$.05/2^k$	51	$4.9222E-04$	$2.4532E-02$
	$.01/2^k$	79	$2.4318E-05$	$4.3945E-03$
	$.001/2^k$	85	$1.4305E-06$	$4.8828E-04$
hat	$.1/2^k$	68	$9.3350E-04$	$2.3346E-02$
	$.05/2^k$	72	$7.1390E-04$	$2.3346E-02$
	$.01/2^k$	102	$5.6258E-05$	$2.9764E-03$
	$.001/2^k$	114	$5.7145E-06$	$5.5308E-04$

TABLE 10
$f_3(x)$ + delta(x-.125) + delta(x-.625)

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
$Eno-cell$	$.2/2^k$	22	$8.4749E-04$	0.1922
	$.02/2^k$	40	$1.9817E-04$	$7.6129E-03$
	$.01/2^k$	50	$4.3982E-05$	$4.5452E-04$
	$.001/2^k$	65	$8.6319E-06$	$4.4052E-05$
$Eno-cell(EC2)$	$.2/2^k$	22	$8.1710E-04$	0.1922
	$.02/2^k$	40	$1.8255E-04$	$7.4825E-03$
	$.01/2^k$	50	$4.4133E-05$	$4.5452E-04$
	$.001/2^k$	65	$8.6312E-06$	$4.4052E-05$
$Eno-hat$	$.1/2^k$	22	$1.6593E-03$	0.1635
	$.02/2^k$	43	$4.1202E-04$	$1.8132E-02$
	$.01/2^k$	53	$4.8946E-05$	$1.2067E-03$
	$.001/2^k$	70	$9.0360E-06$	$2.1094E-04$
$Eno-hat(EC2)$	$.1/2^k$	24	$1.5607E-03$	$7.2525E-02$
	$.02/2^k$	45	$3.8829E-04$	$1.4114E-02$
	$.01/2^k$	53	$4.8939E-05$	$1.2067E-03$
	$.001/2^k$	70	$9.0289E-06$	$2.1094E-04$

TABLE 11
Sine function + delta(x-.125)+delta(x-.5), ENO, r=4

method	*tol*	*nonzeros*	$\|\cdot\|_1 - error$	$\|\cdot\|_\infty - error$
$Eno - cell$	$.4/2^k$	15	$2.4835E-03$	0.1942
	$.2/2^k$	26	$1.3644E-03$	0.1880
	$.1/2^k$	31	$9.1660E-04$	$6.0221E-02$
	$.01/2^k$	50	$3.4917E-04$	$1.5271E-02$
	$.004/2^k$	63	$3.7137E-04$	$6.0973E-02$
	$.001/2^k$	68	$3.5856E-04$	$6.0973E-02$
$Eno - cell(EC2)$	$.4/2^k$	15	$2.3911E-03$	0.1939
	$.2/2^k$	29	$9.8476E-04$	$1.8417E-02$
	$.1/2^k$	31	$6.6734E-04$	$1.0997E-02$
	$.01/2^k$	51	$1.2198E-04$	$3.6628D-03$
	$.004/2^k$	63	$2.4729E-05$	$7.6680E-04$
	$.001/2^k$	68	$1.1865E-05$	$2.3074E-04$
$Eno - hat$	$.4/2^k$	18	$5.1711E-03$	0.4270
	$.2/2^k$	22	$2.6853E-03$	0.1461
	$.1/2^k$	24	$1.9445E-03$	0.1120
	$.01/2^k$	53	$1.9243E-04$	$1.1169E-02$
	$.004/2^k$	57	$6.3767E-05$	$1.6320E-03$
	$.001/2^k$	63	$5.0695E-05$	$9.0668E-04$
$Eno - hat(EC2)$	$.4/2^k$	22	$3.4424E-03$	0.1985
	$.2/2^k$	23	$2.0050E-03$	0.1142
	$.1/2^k$	26	$1.6906E-03$	$7.5666E-02$
	$.01/2^k$	60	$1.4432E-04$	$6.6214E-03$
	$.004/2^k$	57	$5.9641E-05$	$1.6189E-03$
	$.001/2^k$	78	$4.0542E-05$	$6.4321E-04$

TABLE 12

Exponential + delta(x-.125)+delta(x-.5), ENO r=4

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
$Eno - cell$	$.4/2^k$	19	$1.3522E-03$	0.1944
	$.2/2^k$	26	$9.9031E-04$	0.1890
	$.1/2^k$	35	$4.0810E-04$	$1.0214E-02$
	$.04/2^k$	38	$2.0221E-04$	$1.0214E-02$
	$.01/2^k$	49	$5.3793E-05$	$3.3377E-03$
	$.001/2^k$	53	$8.9406E-07$	$3.0517E-04$
$Eno - cell(EC2)$	$.4/2^k$	19	$1.3522E-03$	0.1944
	$.2/2^k$	26	$9.7679E-04$	0.1890
	$.1/2^k$	35	$3.7304E-04$	$1.0214E-02$
	$.04/2^k$	38	$1.8059E-04$	$1.0214E-02$
	$.01/2^k$	49	$2.4163E-05$	$3.3377E-03$
	$.001/2^k$	53	$8.9406E-07$	$3.0517E-04$
$Eno - hat$	$.4/2^k$	29	$3.0716E-03$	0.3930
	$.2/2^k$	34	$1.0038E-03$	0.1950
	$.1/2^k$	36	$5.8534E-04$	$9.1972E-02$
	$.04/2^k$	41	$3.2235E-04$	$4.6238E-02$
	$.01/2^k$	49	$4.2495E-05$	$1.0245E-02$
	$.001/2^k$	55	$2.5033E-06$	$4.2724E-04$
$Eno - hat(EC2)$	$.4/2^k$	31	$3.2741E-03$	0.1893
	$.2/2^k$	36	$1.1155E-03$	$9.1972E-02$
	$.1/2^k$	38	$6.0777E-04$	$6.6395E-02$
	$.04/2^k$	42	$3.2724E-04$	$2.0851E-02$
	$.01/2^k$	51	$4.0581E-05$	$5.2556E-03$
	$.001/2^k$	55	$2.5033E-06$	$4.2724E-04$

TABLE 13

$f_3(x)$ + delta(x-.125)+delta(x-.5), ENO r=4

method	*tol*	*nonzeros*	$\|\cdot\|_1$ *−error*	$\|\cdot\|_\infty$ *−error*
$Eno-cell$	$.4/2^k$	7	$1.2758E-03$	0.1944
	$.2/2^k$	21	$8.2509E-04$	$1.1051E-02$
	$.1/2^k$	24	$4.5058E-04$	$9.0340E-03$
	$.04/2^k$	29	$2.5545E-05$	$7.5659E-03$
	$.01/2^k$	38	$4.8189E-05$	$1.0596E-03$
	$.001/2^k$	49	$2.0243E-06$	$2.0464E-05$
$Eno-cell(EC2)$	$.4/2^k$	6	$1.0488E-03$	0.1945
	$.2/2^k$	21	$7.3934E-04$	$9.9057E-03$
	$.1/2^k$	24	$3.9894E-04$	$8.4899E-03$
	$.04/2^k$	29	$2.3187E-04$	$7.4425E-03$
	$.01/2^k$	38	$4.8155E-05$	$1.0596E-03$
	$.001/2^k$	49	$2.0243E-06$	$2.0464E-05$
$Eno-hat$	$.4/2^k$	21	$1.5399E-03$	0.1936
	$.2/2^k$	28	$1.0135E-03$	0.1599
	$.1/2^k$	31	$9.2890E-04$	0.1233
	$.04/2^k$	43	$2.7176E-04$	$1.8225E-02$
	$.01/2^k$	51	$3.2673E-05$	$1.1349E-03$
	$.001/2^k$	60	$2.8879E-06$	$7.4028E-05$
$Eno-hat(EC2)$	$.4/2^k$	21	$1.5399E-03$	0.1936
	$.2/2^k$	23	$1.1502E-03$	0.1889
	$.1/2^k$	31	$7.8246E-04$	$4.3128E-02$
	$.04/2^k$	44	$3.1718E-04$	$1.0132E-02$
	$.01/2^k$	51	$3.2505E-05$	$1.1837E-03$
	$.001/2^k$	60	$2.8134E-06$	$8.2333E-05$

TABLE 14

Discontinuous sine + delta(x-.125)+delta(x-.625), ENO r=4

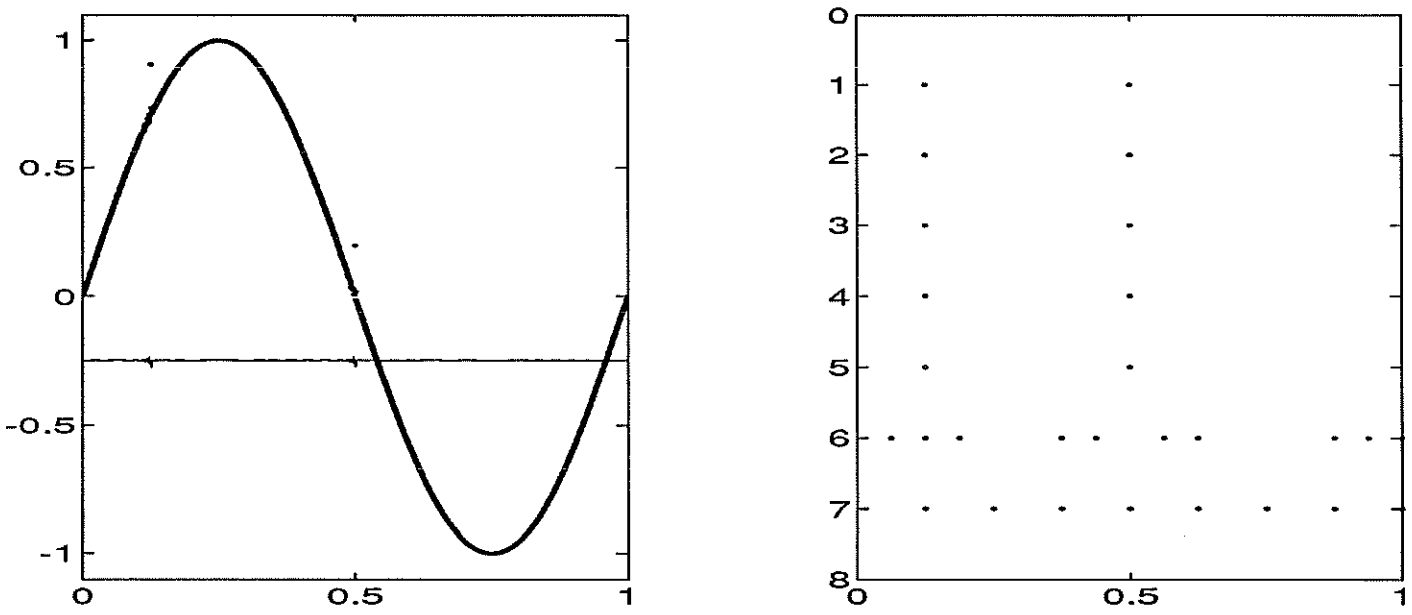

FIG. 1. *Cell-average. tol=.1, nz=86, l1=6.99E-04, sup=2.44E-02*

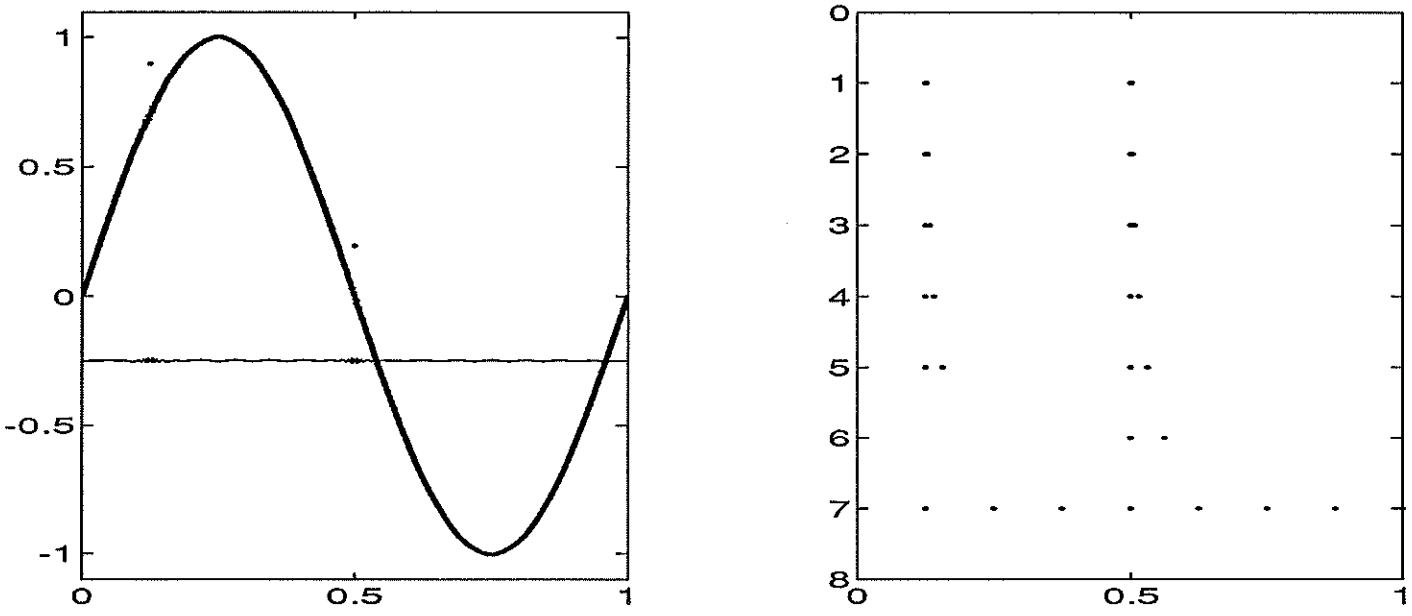

FIG. 2. *Hat-weighted. tol=.1, nz=87, l1=1.17E-04, sup=1.04E-02*

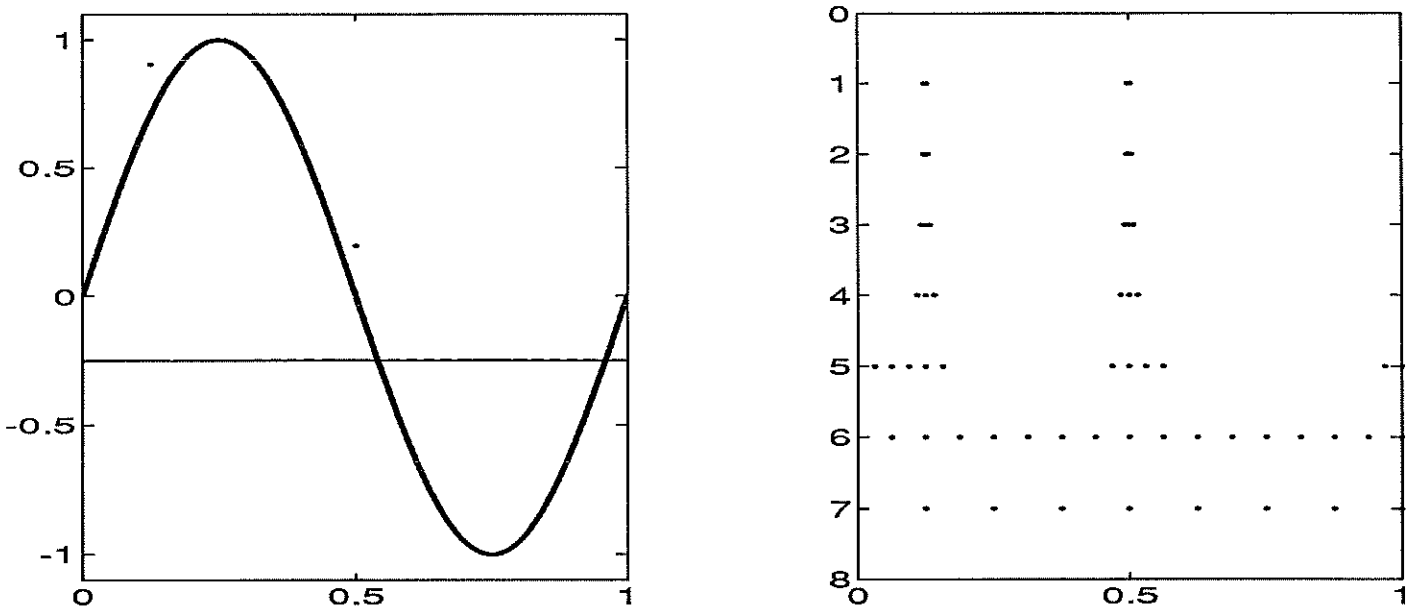

FIG. 3. *Cell-average. tol=.01, nz=67, l1=1.23E-04, sup=4.34E-04*

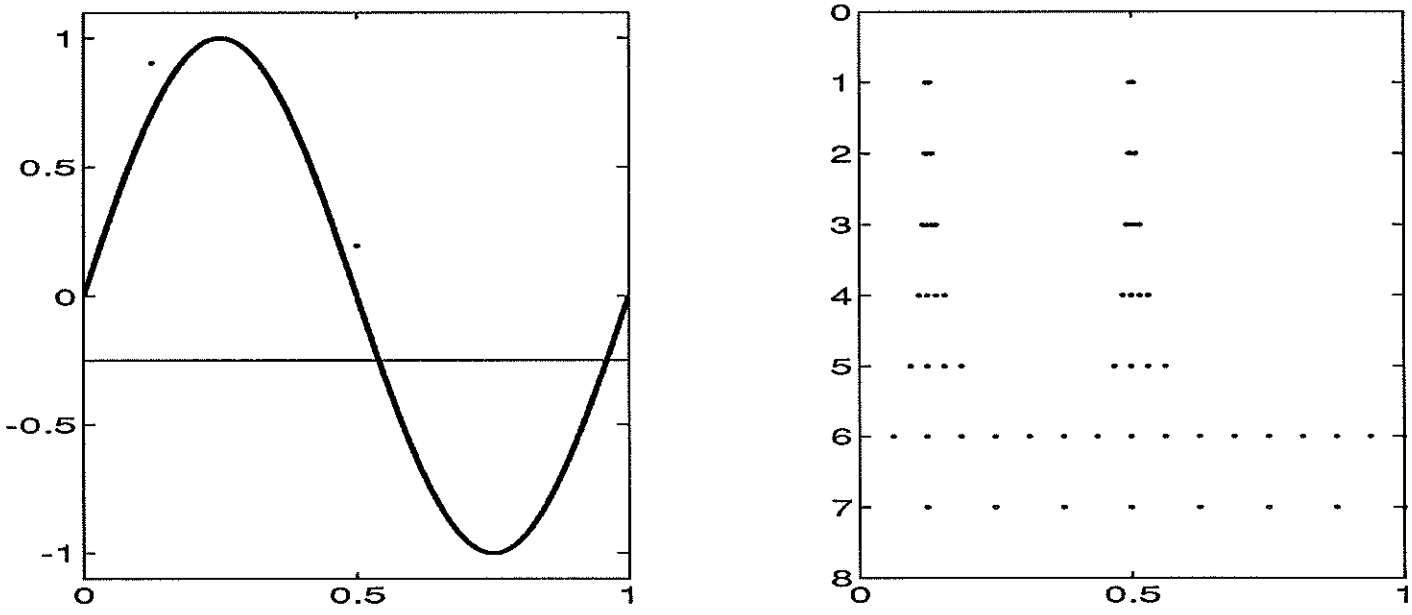

FIG. 4. *Hat-weighted. tol=.01, nz=72, l1=4.46E-05, sup=1.96E-04*

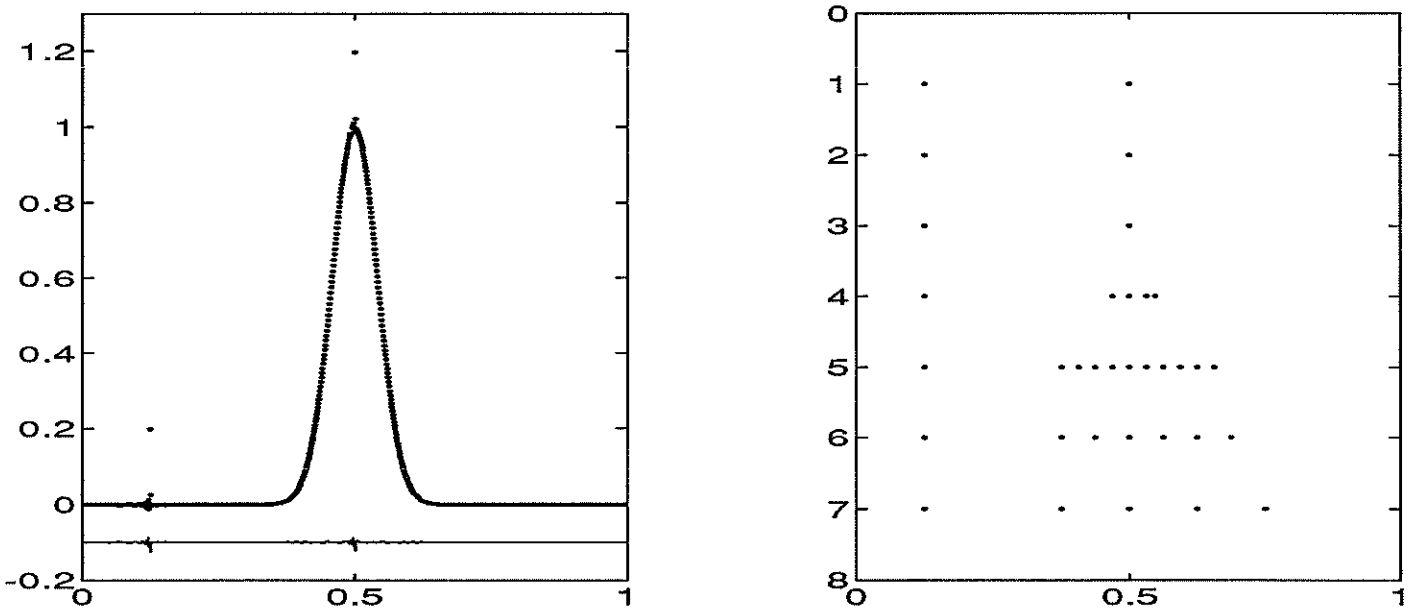

FIG. 5. *Cell-average. tol=.05, nz=42, l1=5.79E-04, sup=2.46E-02*

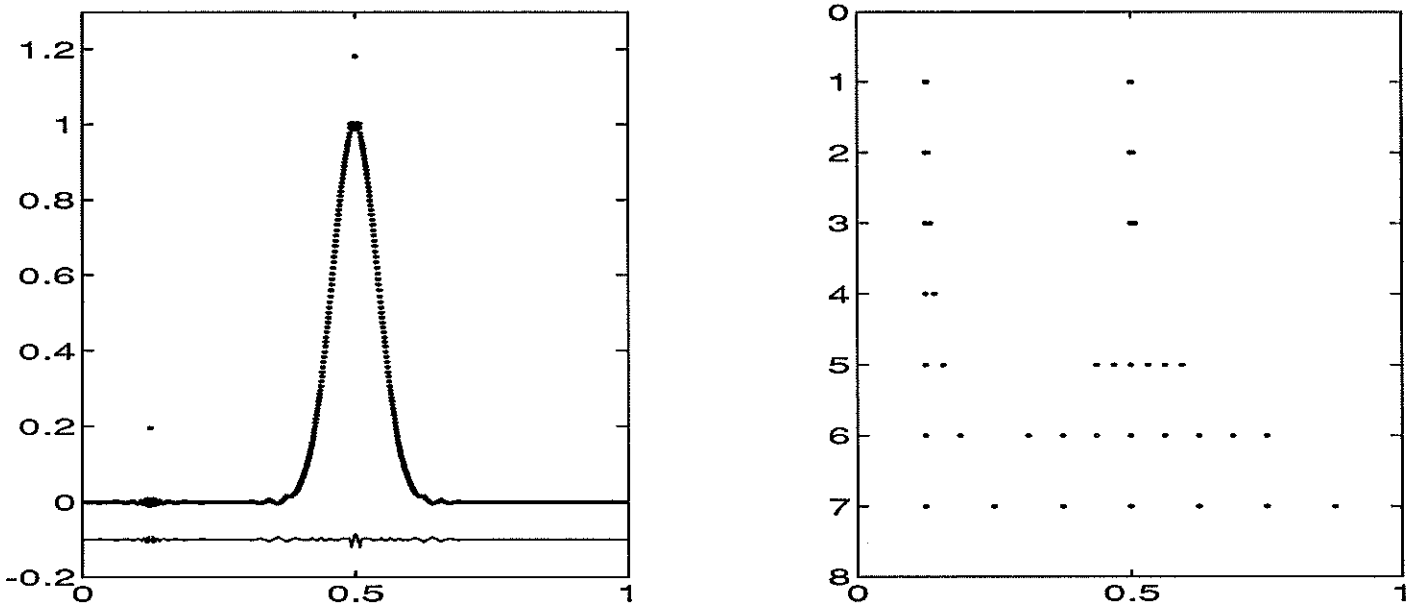

FIG. 6. *Hat-weighted. tol=.1, nz=47, l1=1.12E-04, sup=2.07E-02*

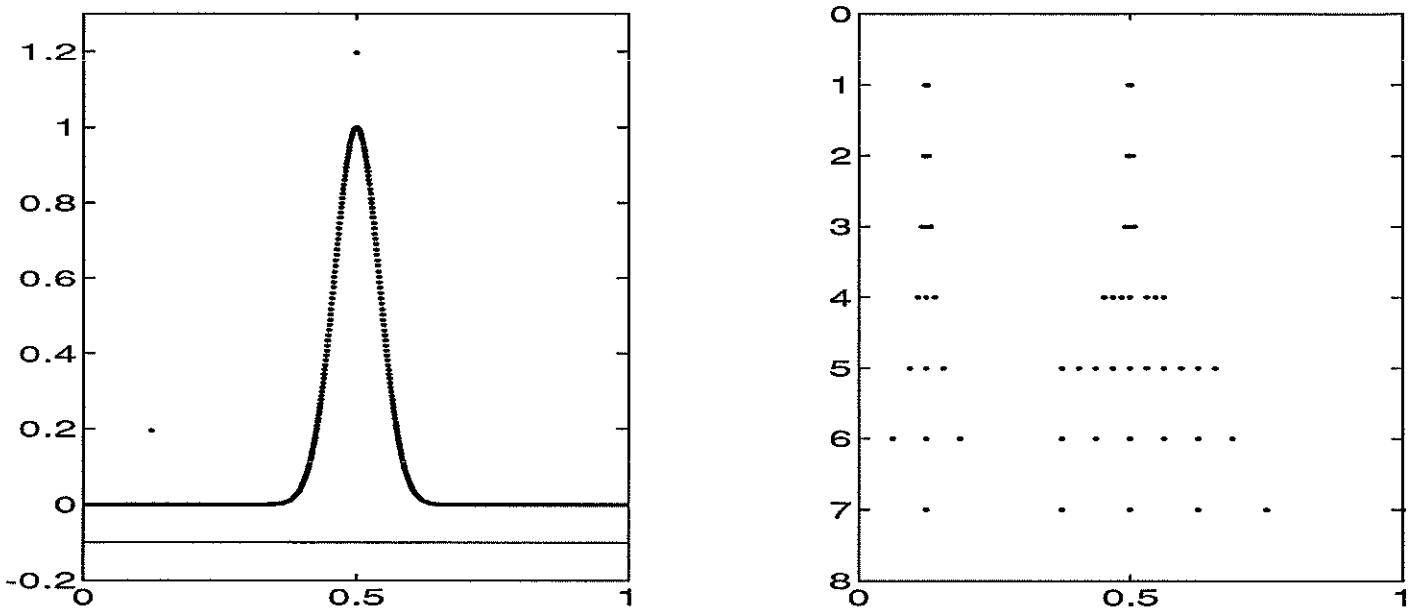

FIG. 7. *Cell-average. tol=.02, nz=64, l1=1.94E-04, sup=1.26E-03*

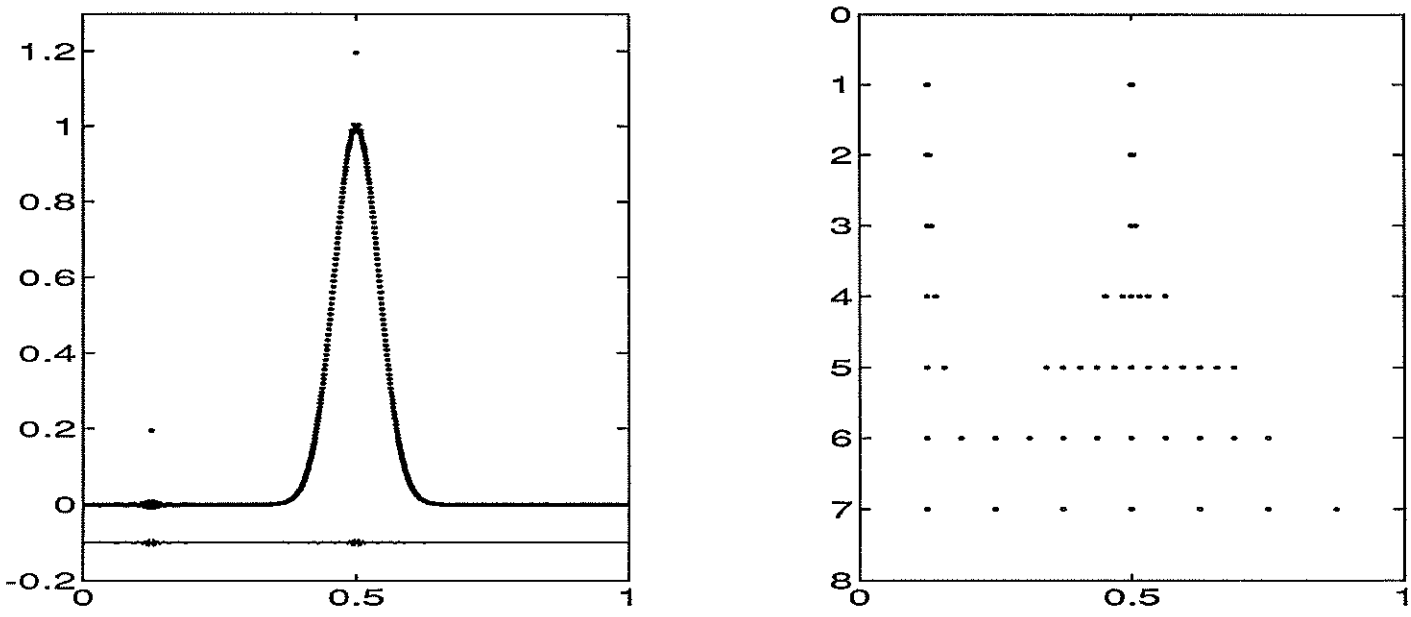

FIG. 8. *Hat-weighted. tol=.02, nz=60, l1=5.15E-04, sup=8.42E-03*

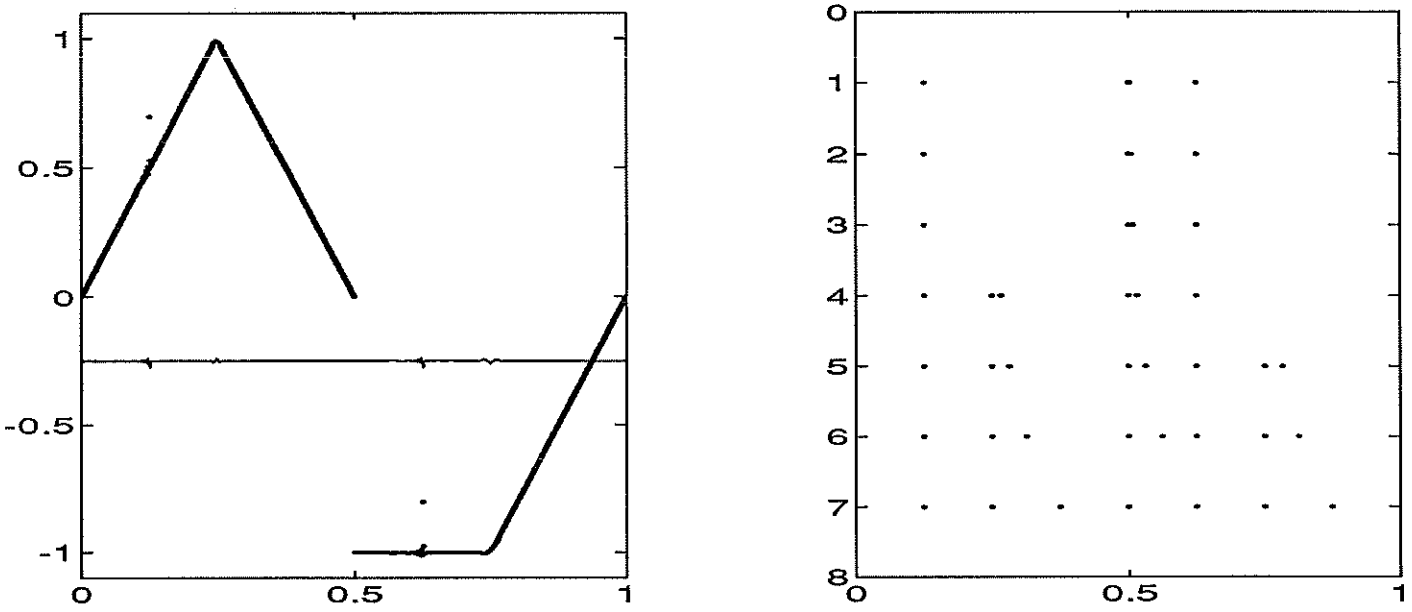

FIG. 9. *Cell-average. tol=.1, nz=49, l1=5.93E-04, sup=2.45E-02*

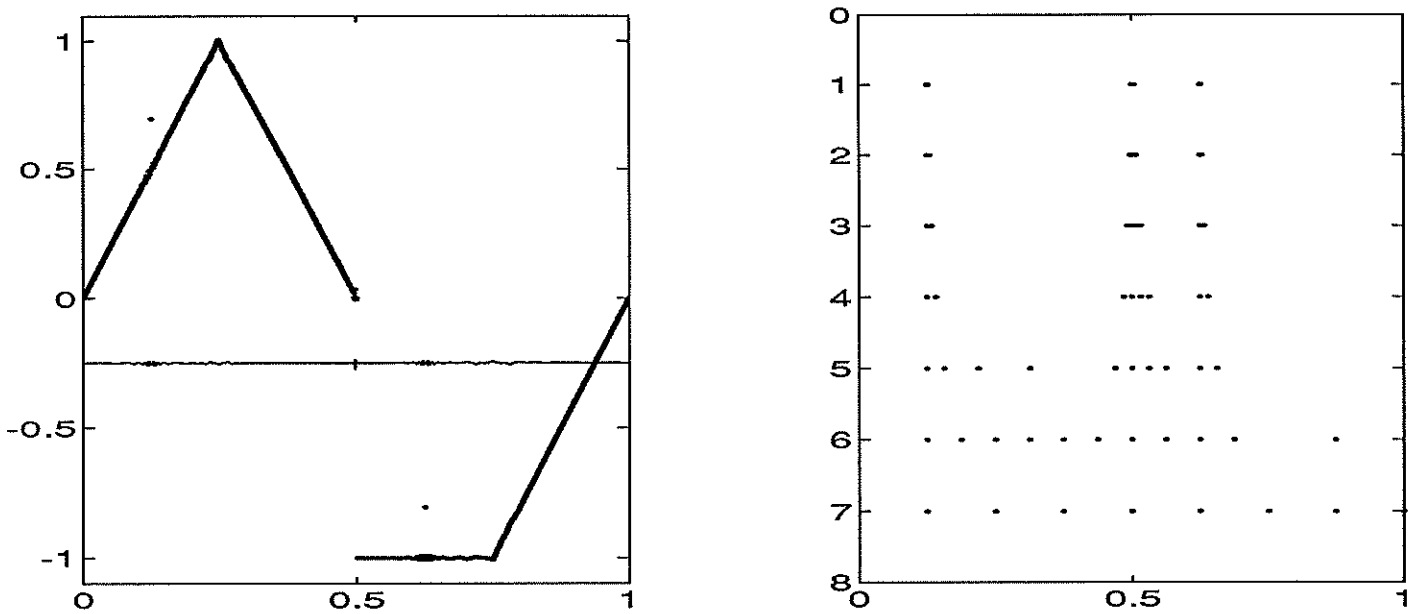

FIG. 10. *Hat-weighted. tol=.1, nz=68, l1=9.35E-04, sup=2.33E-02*

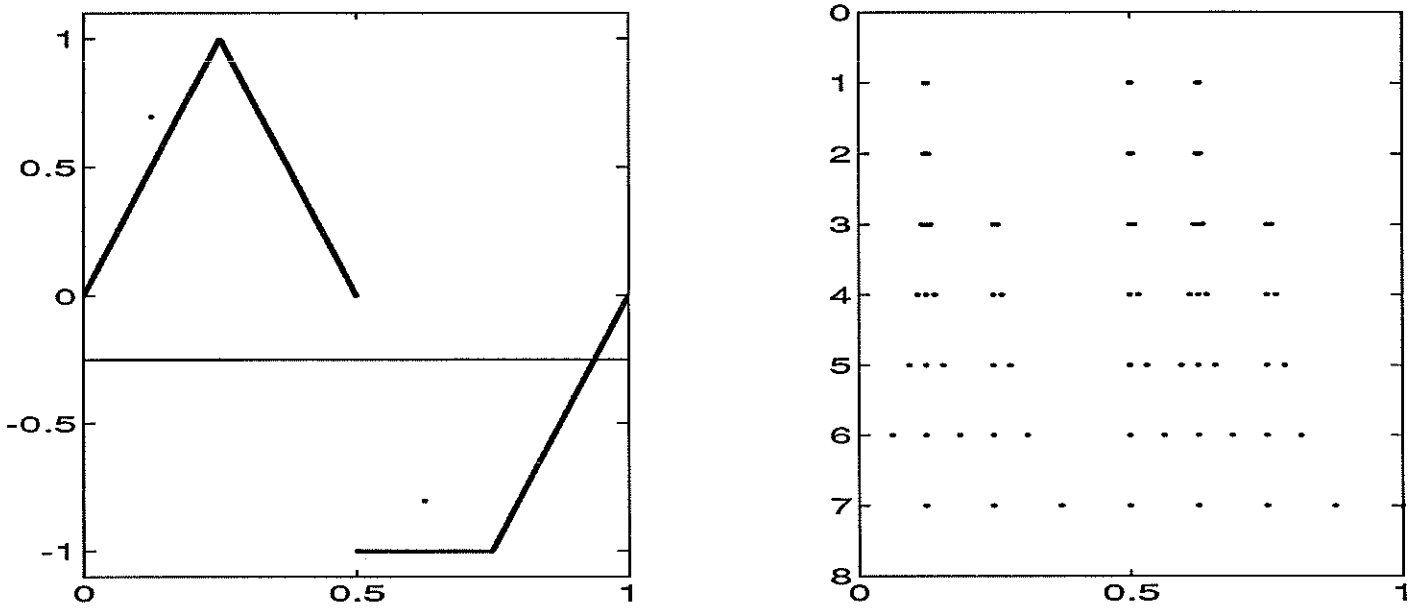

FIG. 11. *Cell-average. tol=.01, nz=79, l1=2.43E-05, sup=4.39E-03*

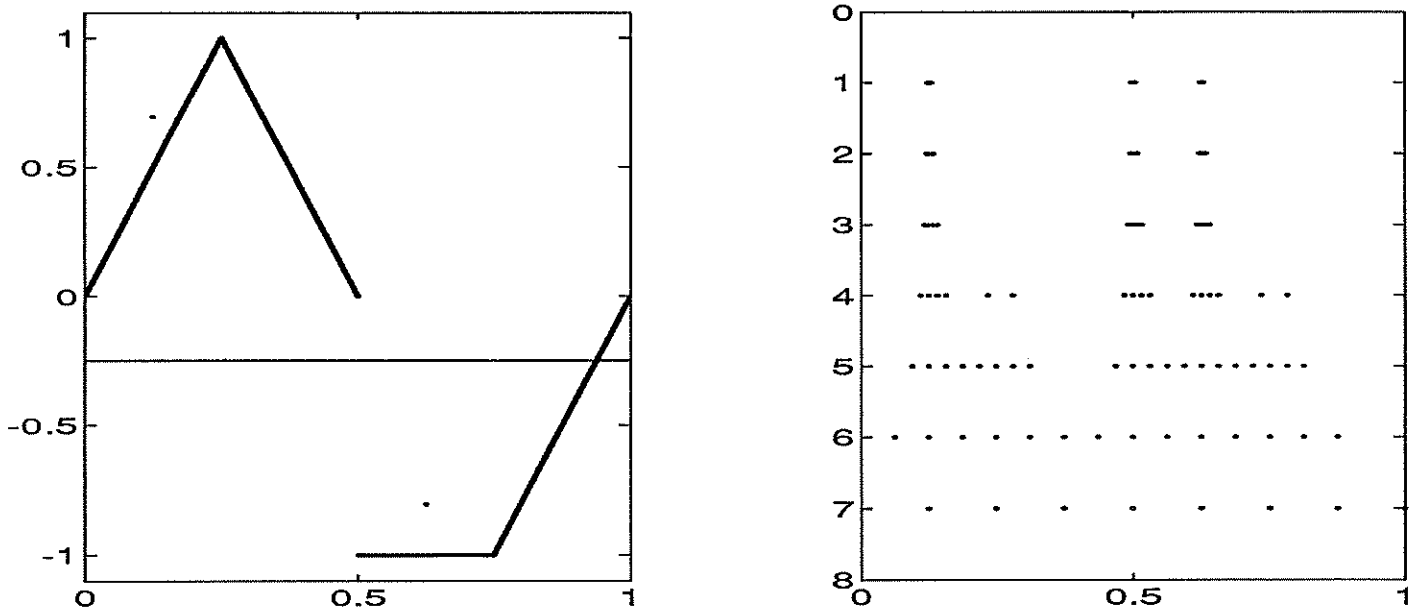

FIG. 12. *Hat-weighted. tol=.01, nz=102, l1=5.62E-05, sup=2.97E-03*

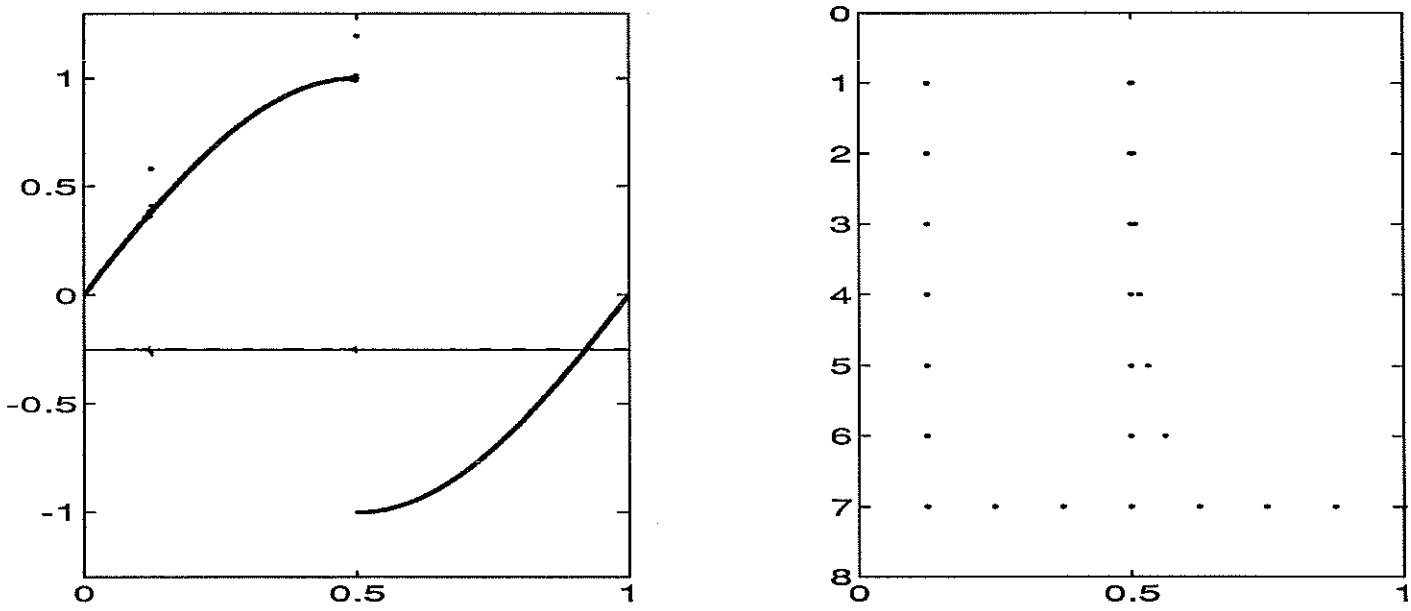

FIG. 13. *Cell-average. tol=.1, nz=34, l1=4.62E-04, sup=2.48E-02*

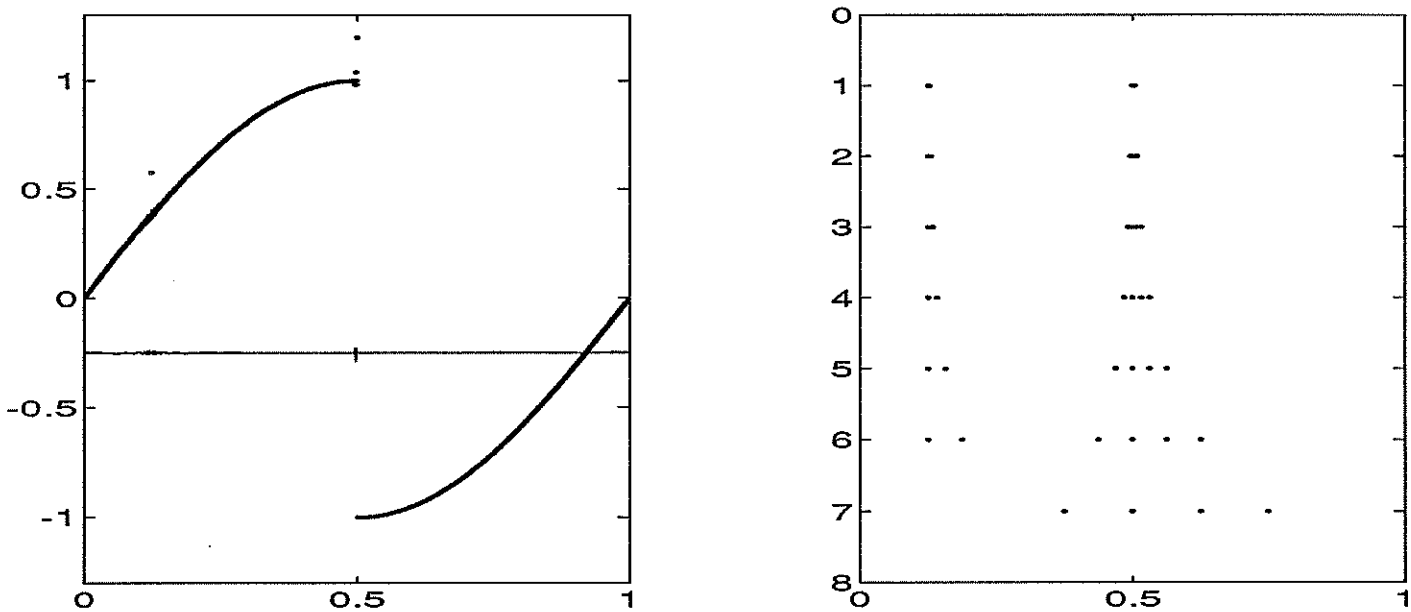

FIG. 14. *Hat-weighted. tol=.1, nz=47, l1=5.34E-04, sup=3.77E-02*

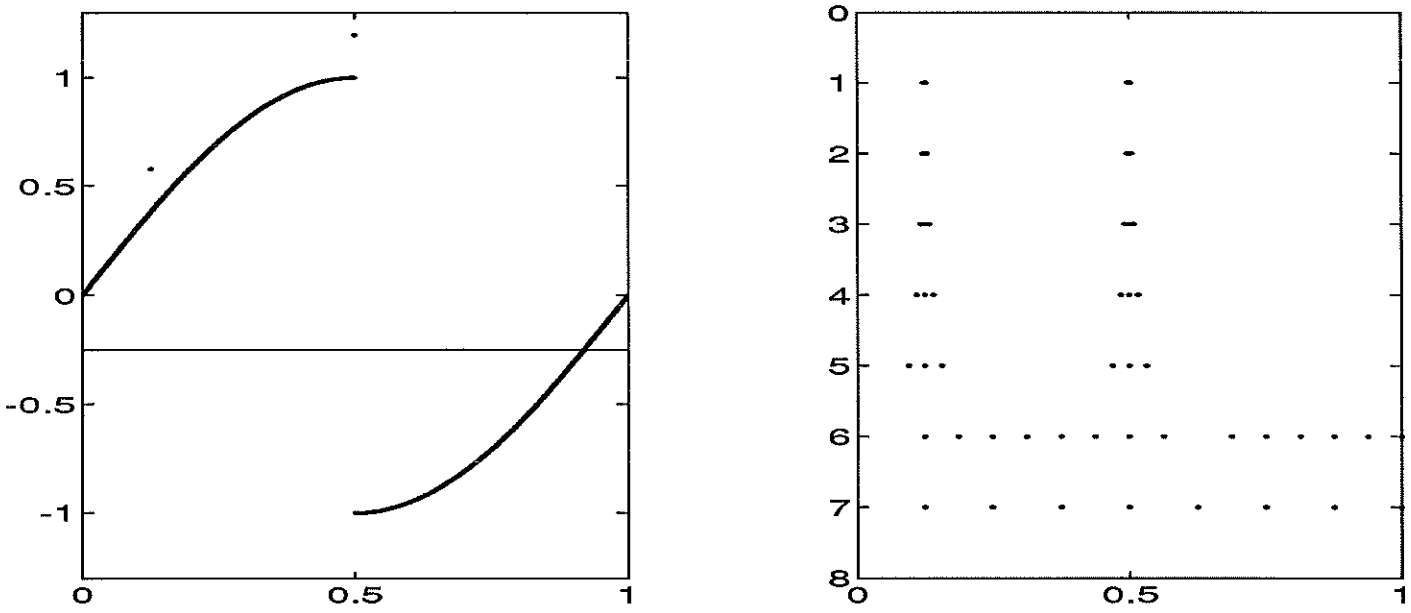

FIG. 15. *Cell-average. tol=.01, nz=60, l1=2.99E-05, sup=1.44E-04*

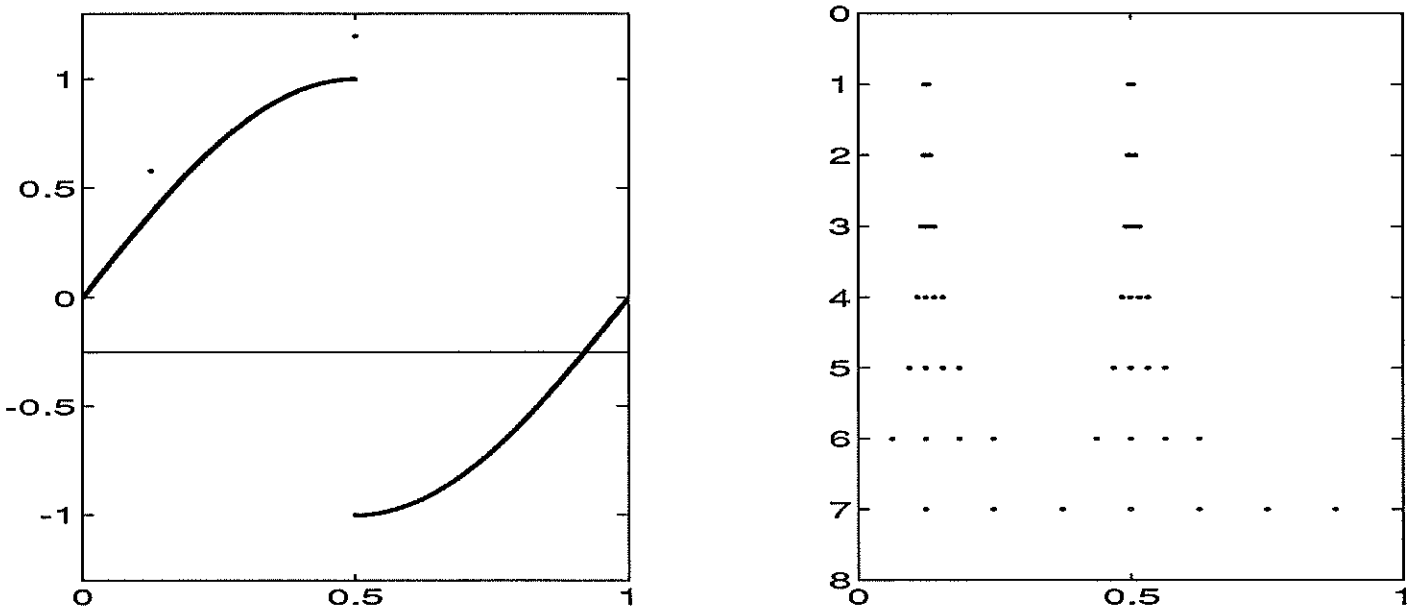

FIG. 16. *Hat-weighted. tol=.01, nz=63, l1=3.02E-05, sup=1.66E-04*

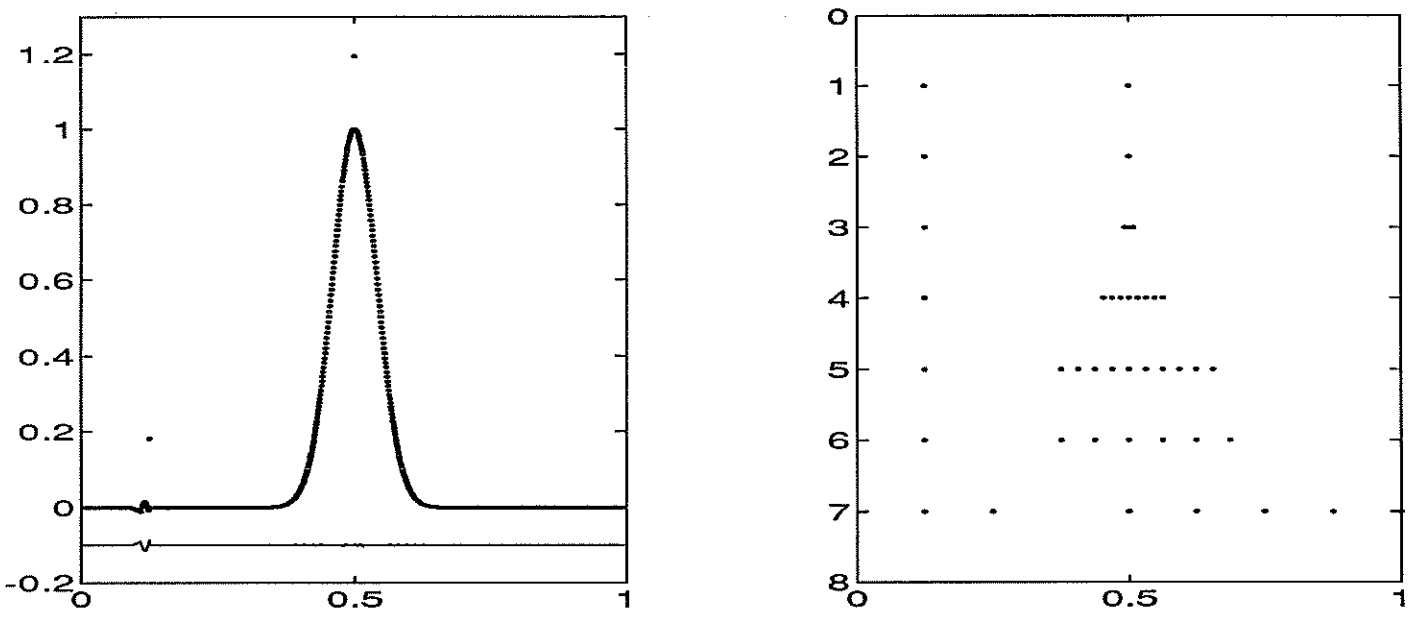

FIG. 17. *Cell-ENO. tol=.01, nz=50, l1=3.49E-04, sup=1.52E-02*

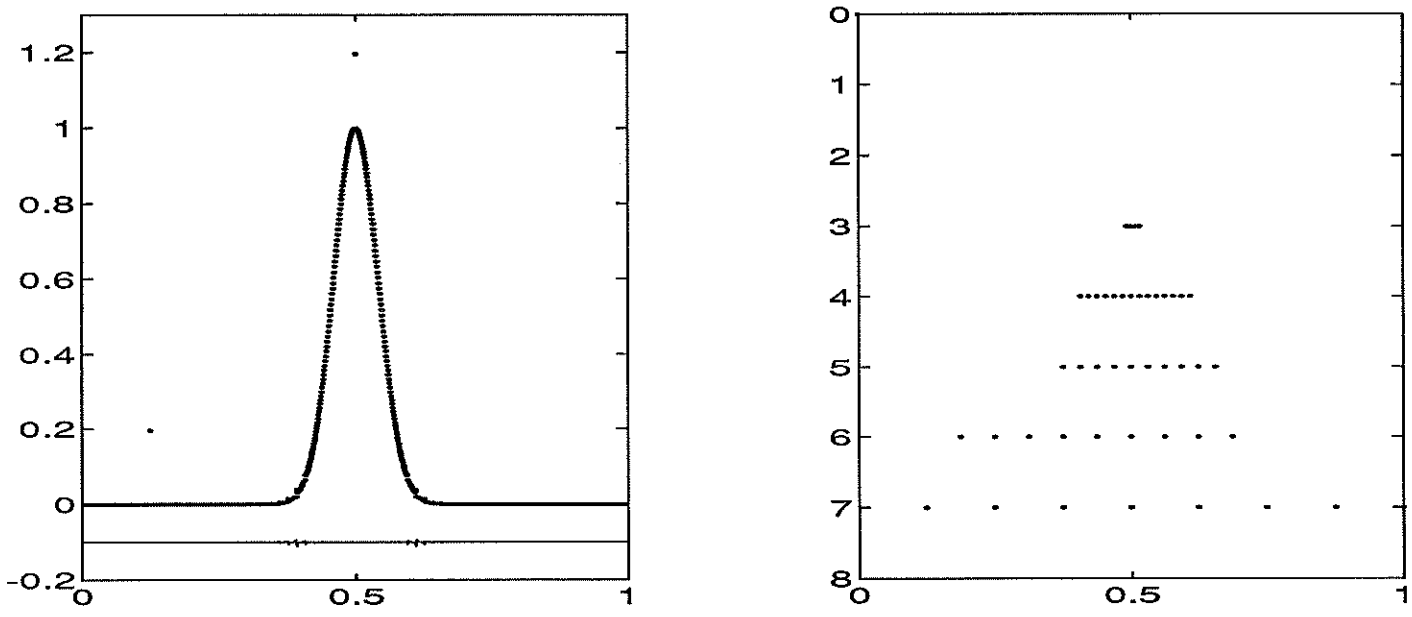

FIG. 18. *Hat-ENO. tol=.01, nz=53, l1=1.92E-04, sup=1.11E-02*

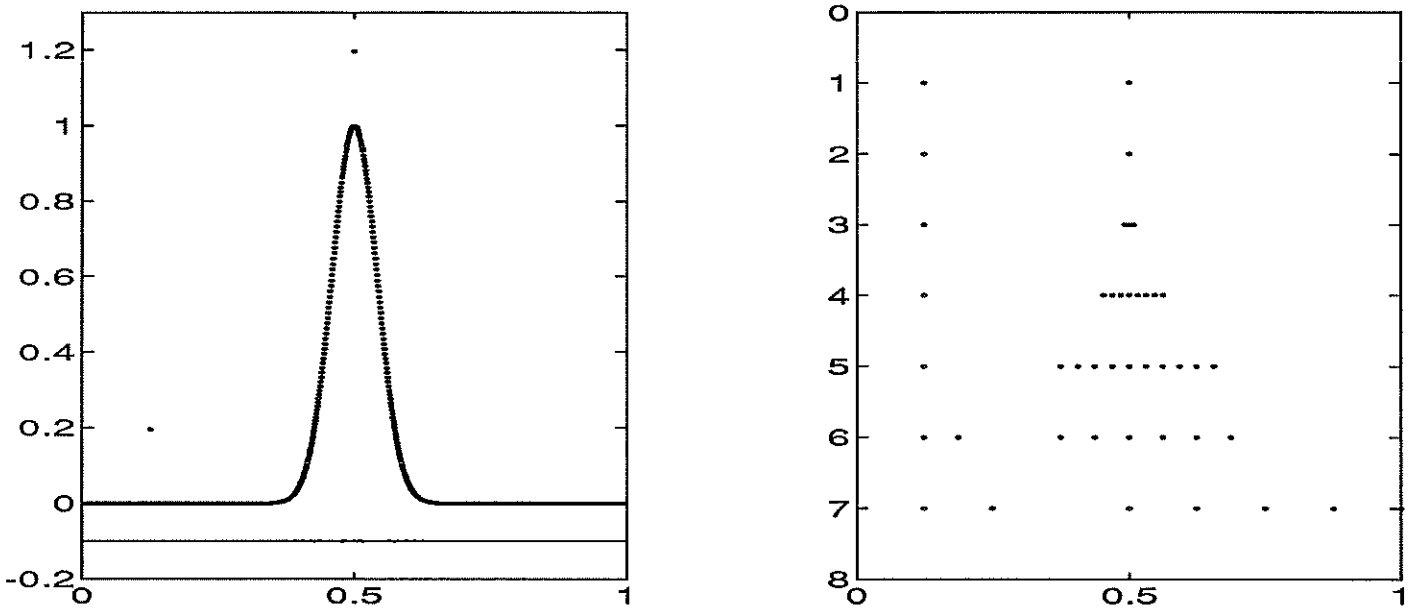

FIG. 19. *Cell-ENO EC. tol=.01, nz=51, l1=1.21E-04, sup=3.66E-03*

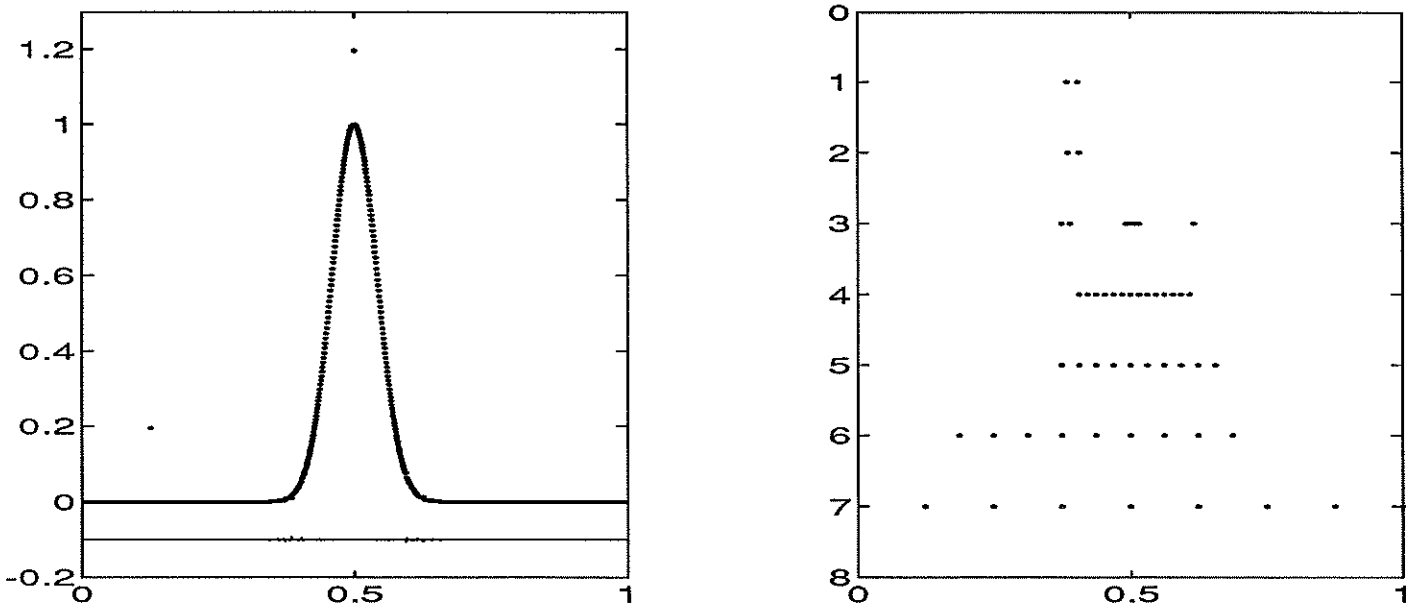

FIG. 20. *Hat-ENO EC. tol=.01, nz=60, l1=1.44E-04, sup=6.62E-03*

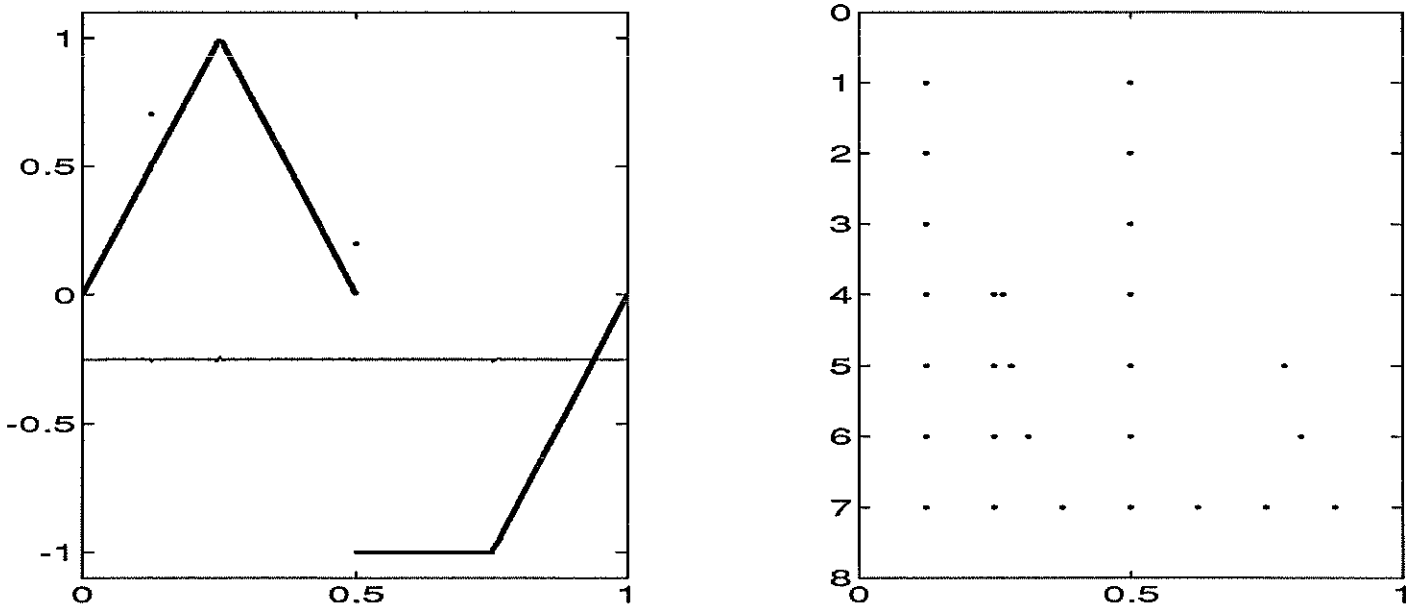

FIG. 21. *Cell-ENO. tol=.1, nz=35, l1=4.08E-04, sup=1.02E-02*

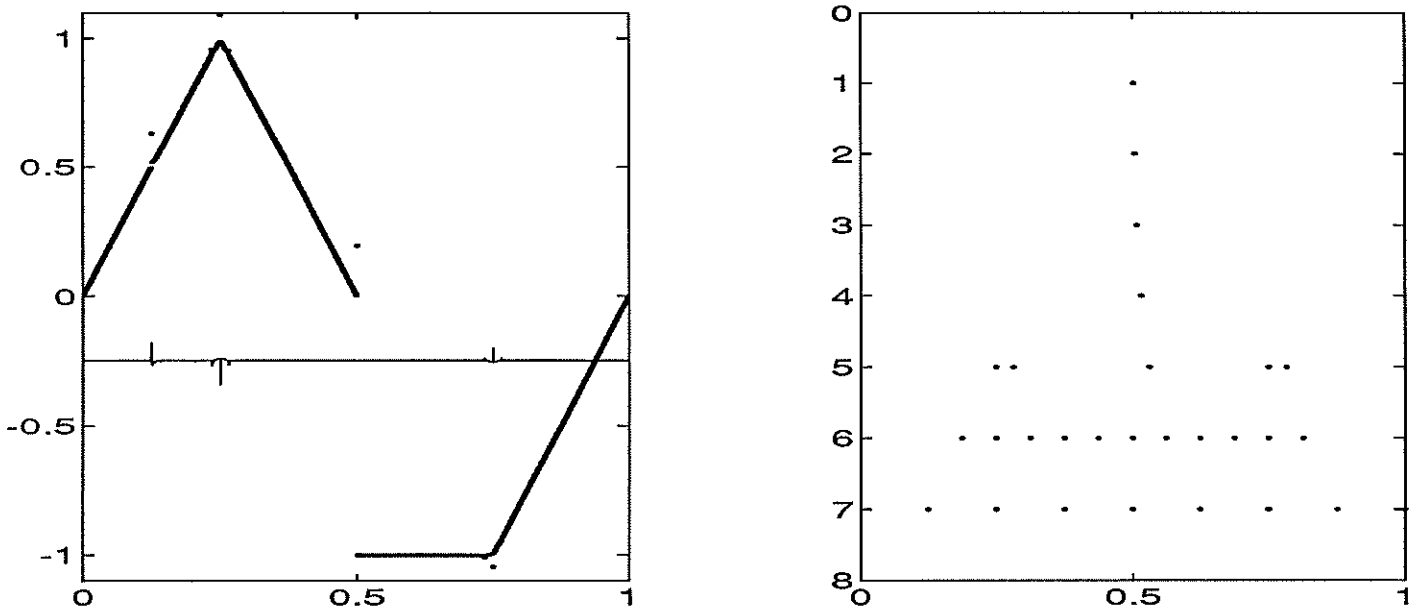

FIG. 22. *Hat-ENO. tol=.1, nz=36, l1=5.85E-04, sup=9.19E-02*

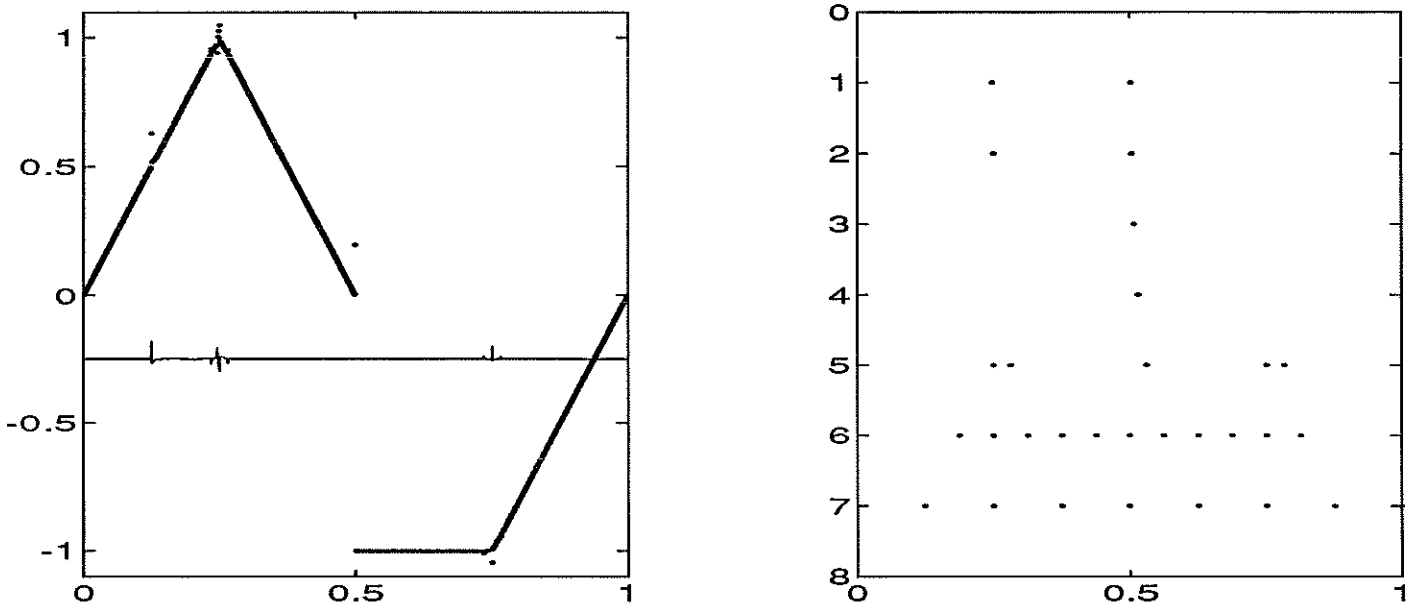

FIG. 23. *Hat-ENO EC. tol=.1, nz=38, l1=6.07E-04, sup=6.63E-02*

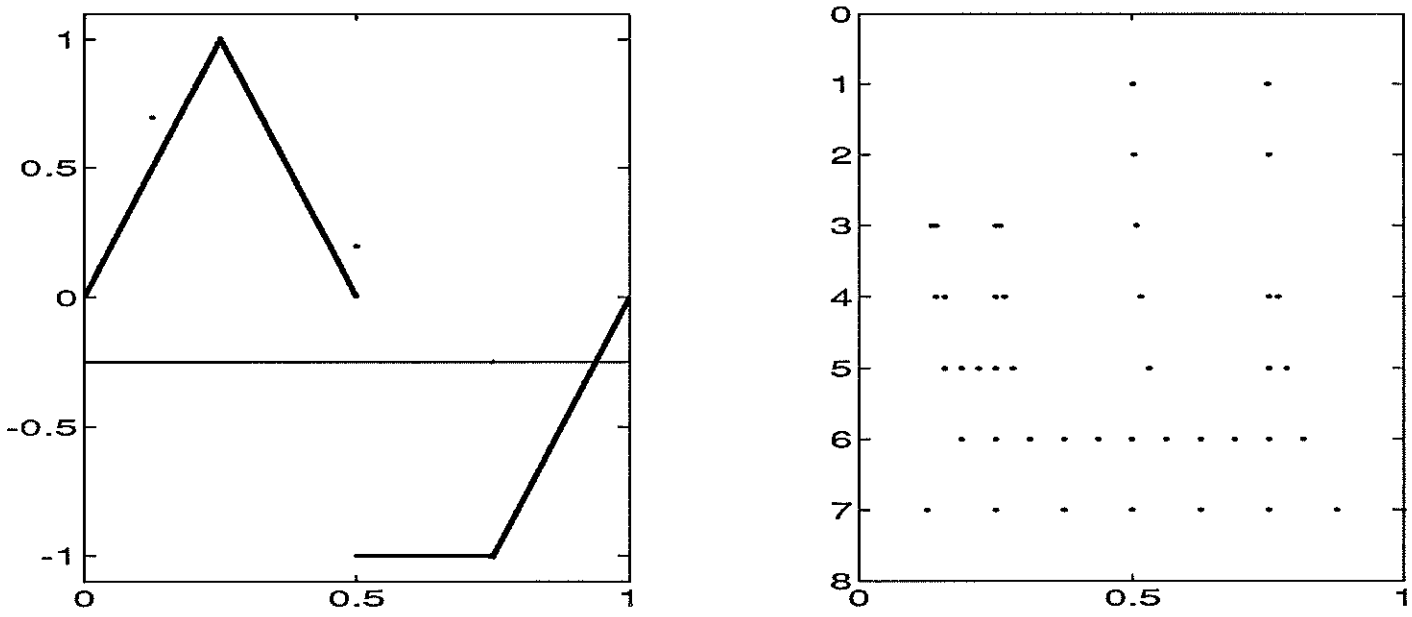

FIG. 24. *Hat-ENO EC. tol=.01, nz=51, l1=4.05E-05, sup=5.25E-03*

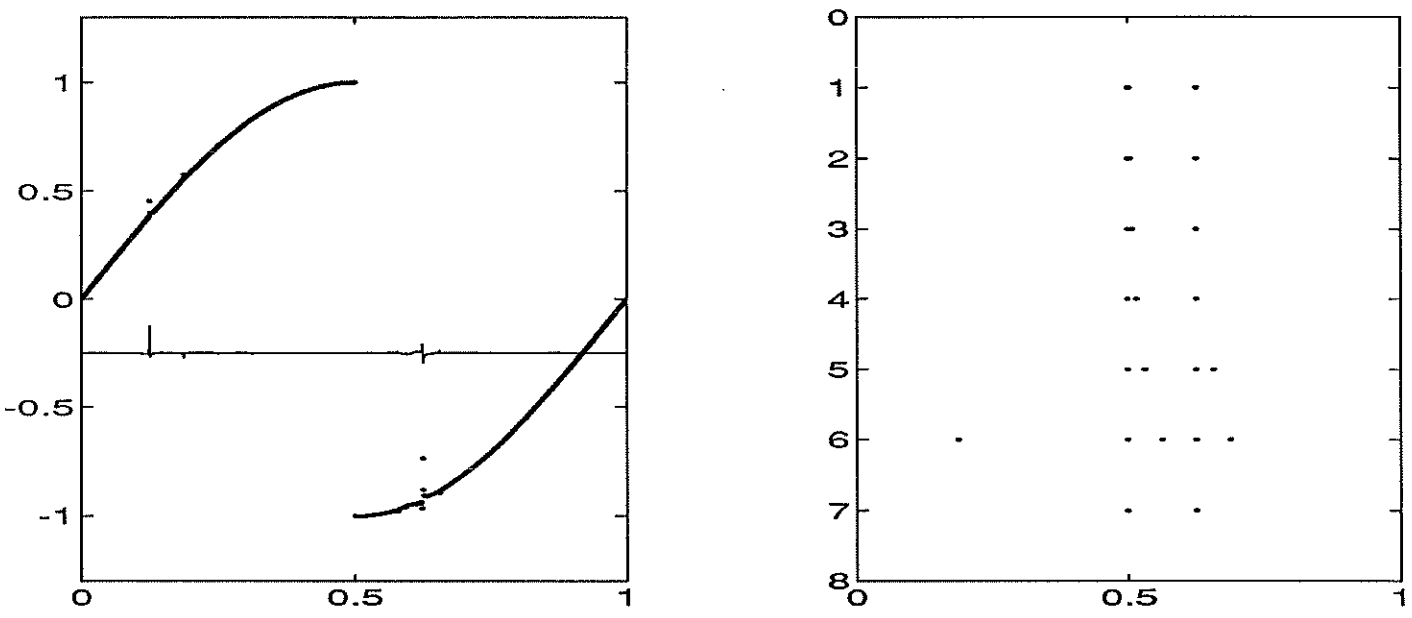

FIG. 25. *Hat-ENO tol=.1, nz=31, l1=9.28E-04, sup=0.123*

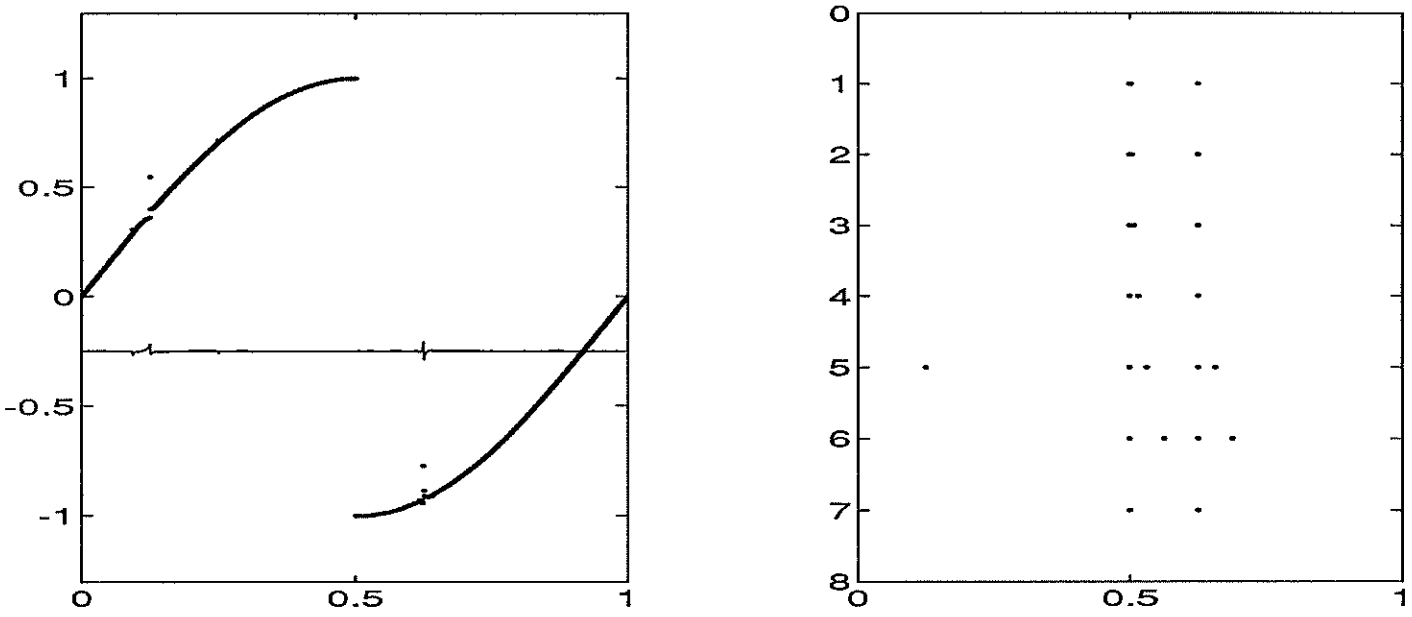

FIG. 26. *Hat-ENO EC. tol=.1, nz=51, l1=7.82E-04, sup=4.31E-02*

REFERENCES

[1] F. Aràndiga and V. Candela, *Multiresolution Standard Form of a Matrix*, UCLA Computational and Applied Mathematics Report 92-37 (1992)

[2] F. Aràndiga, V. Candela and R. Donat, *Fast Multiresolution Algorithms for Solving Linear Equations: A Comparative Study* , UCLA Computational and Applied Mathematics Report 92-52 (1992)

[3] E. Bacry, S. Mallat and G. Papanicolau, *A Wavelet based space-time adaptive numerical method for partial differentia equations* Preprint, To appear in Matehmatical MOdeling and Nuemrical Analysis, 1993

[4] G. Beylkin, *Wavelets, Multiresolution Analysis and Fast Numerical Algorithms*, preprint

[5] G. Beylkin, R. Coifman and V. Rokhlin, *Fast Wavelet Transform and Numerical Algorithms I.*, Comm. Pure Appl. Math., **XLIV**, pp. 141-183, (1991)

[6] G. Deslauriers, S. Duboc, *Symmetric iterative interpolation scheme*, Constructive Approximation **5** 49-68 (1989)

[7] R. Donat, A. Harten, *Data Compression Algorithms for Locally Oscilatory Data* UCLA Computational and Applied Mathematics Report 93— (1993)

[8] N. Dyn, J.A. Gregory, D. Levin *Analysis of Linear binary subdivision schemes for curve design*, Constructive Approximation **7** 127-147 (1991)

[9] B. Engquist, S. Osher, and S. Zhong, *Fast Wavelet Algorithms for Linear Evolution Equations*, ICASE Report 92-14 (1992)

[10] A. Harten, *ENO schemes with Subcell Resolution*, J. Comp. Physics, v. 83 (1989), pp. 148-184; also ICASE Report # 87-56

[11] A. Harten, *Discrete Multiresolution Analysis and Generalized Wavelets*, UCLA Computational and Applied Mathematics Report 92-08 (1992)

[12] A. Harten and I. Yad-Shalom *Fast Multiresolution Algorithms for Matrix-Vector Multiplication*, UCLA Computational and Applied Mathematics Report 92-31 (1992)

[13] A. Harten, *Multiresolution Representation of Data*, UCLA Computational and Applied Mathematics Report 93-13 (1993)

[14] A. Harten, *Multiresolution Algorithms for the Numerical Solution of Hyperbolic Conservation Laws*, UCLA Computational and Applied Mathematics Report 93-03 (1993)

[15] A. Harten, *Adaptive Multiresolution Schemes for Shock Computations*, UCLA Computational and Applied Mathematics Report 93-06 (1993)

Appendix A

Both when we have to obtain the coefficients of the spline interpolation in section (6.2) and when we have to calculate the elements e_i, $1 \leq i \leq 2n$, from d_i, $1 \leq i \leq n$, in the decoding algorithm of the orthogonal case in 5.2, we need to solve a linear system of equations $Ax = b$, where A is a $n \times n$ matrix of the form

$$\begin{bmatrix} a & b & 0 & \dots & \dots & 0 & b \\ b & a & b & 0 & \dots & \dots & 0 \\ 0 & b & \ddots & \ddots & 0 & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & \\ & & & & & \ddots & 0 \\ 0 & & & & \ddots & \ddots & b \\ b & 0 & \dots & & 0 & b & a \end{bmatrix} \tag{154}$$

If A were tridiagonal, we know we can solve the system with $\mathcal{O}(n)$ operations. Our goal, in this appendix, is to show that we can solve the systems (93) and (123), where A is a tridiagonal "periodic" matrix, also in $\mathcal{O}(n)$ operations.

Let us denote by A_k the $k \times k$ submatrix formed by the first k rows and columns in A. If we assume that $|a| > 2|b|$, which is the case in (93) and (123), we have that each A_k is strictly diagonally dominant, hence invertible and

$$det(A_k) \neq 0. \tag{155}$$

This guarantees that the LU decomposition of A exists. Moreover, all the eigenvalues of A are positive, hence A is positive definite. Therefore Gaussian elimination without row or column interchanges is stable and it can be used to compute the matrices L and U.

Due to the structure of A, the matrices L and U have the following form:

$$L = \begin{bmatrix} 1 & 0 & \dots & & \dots & 0 & 0 \\ m_{1,2} & 1 & 0 & \dots & & \dots & 0 \\ 0 & m_{3,2} & \ddots & \ddots & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & \\ & & & \ddots & \ddots & \ddots & \\ 0 & 0 & \dots & 0 & m_{n-1,n-2} & 1 & 0 \\ m_{n,1} & m_{n,2} & \dots & & m_{n,n-2} & m_{n,n-1} & 1 \end{bmatrix}; \tag{156}$$

and

$$U = \begin{bmatrix} u_{1,1} & u_{1,2} & 0 & \dots & \dots & 0 & u_{1,n} \\ 0 & u_{2,2} & u_{2,3} & 0 & \dots & 0 & u_{2,n} \\ \vdots & 0 & \ddots & \ddots & \ddots & \vdots & \vdots \\ & & \ddots & \ddots & & 0 & \\ & & & & \ddots & \ddots & \vdots \\ 0 & & & & \ddots & u_{n-1,n-1} & u_{n-1,n} \\ 0 & 0 & \dots & & \dots & 0 & u_{n,n} \end{bmatrix}. \tag{157}$$

We can calculate L and U, in $\mathcal{O}(n)$ operations, with the following algorithm:

$$\begin{cases} u_{1,1} = a \\ \begin{cases} \text{Do } i = 1, n-2 \\ u_{i,i+1} = b \\ m_{i+1,i} = b/u_{i,i} \\ ui+1, i+1 = a - m_{i+1,i} * b \end{cases} \\ m_{n,1} = b/u_{1,1} \\ u_{1,n} = b \\ \begin{cases} \text{Do } i = 2, n-2 \\ m_{n,i} = -m_{n,i-1} * b/u_{i,i} \\ u_{i,n} = -m_{i,i-1} * u_{i-1,n} \end{cases} \\ m_{n,n-1} = b/u_{n-1,n-1} - m_{n,n-2} * b/u_{n-1,n-1} \\ u_{n-1,n} = b - m_{n-1,n-2} * u_{n-2,n} \\ sum = 0 \\ \begin{cases} \text{Do } i = 1, n-1 \\ sum = sum + m_{n,i} * u_{i,n} \end{cases} \\ u_{n,n} = a - sum. \end{cases} \tag{158}$$

The system $Ax = b$ is equivalent to $LUx = b$, which decompose into two triangular systems

$$Ly = b \quad \text{and } Ux = y$$

which can be solved in $\mathcal{O}(n)$ operations, from

$$\begin{cases} y_1 = b_1 \\ y_i = b_i - m_{i+1,i} * y_{i-1} \quad i = 2, \dots, n-1 \\ y_n = b_n - m_{n,1} * y_1 - m_{n,2} * y_2 - \dots - m_{n,n-1} * y_{n-1} \end{cases} \tag{159}$$

and

$$\begin{cases} x_n = y_n/u_{1,1} \\ x_{n-1} = (y_{n-1} - x_n * u_{n-1,n}/u_{n,n} \\ x_i = (y_i - x_n * u_{i,n} - x_{i+1} * u_{i,i+1})/u_{i,i} \quad i = n-2, \dots, 1 \end{cases} \tag{160}$$

Appendix B

In this appendix, we use the interpolatory results of [6] and [8] to prove convergence of the limiting process (121) under appropriate smoothness assumptions.

The symmetric interpolatory procedures (and corresponding reconstructions) are both translation invariant and the same for each level of resolution, these two facts, together with 120, allow us to study the limiting process by considering the case of infinite grids in the real line. Hence, we consider the decimation and prediction operators to be infinite Toeplitz matrices (see [13]), and the same for each level of resolution. Thus $A_0^m = \prod_{k=0}^{m-1} P_k^{k+1} = P^m$

Let us study the convergence properties of the sequence $P^m\delta^0$ as $m \to +\infty$, where

$$\delta_j^0 = \begin{cases} 1 & j = 0 \\ 0 & j \neq 0 \end{cases} \tag{161}$$

Denote by $\tilde{P}$ the prediction matrix for the interpolatory procedure. Deslauries and Duboc ([6]) and Dyn, Gregory and Levin [8] have studied this limiting process. They have shown that the sequence $\tilde{P}^m\delta^0$ converges in the following sense:

$$\lim_{m\to+\infty} \sum_{j=-\infty}^{+\infty} (\tilde{P}^m\delta^0)_j \chi_{[\frac{j-1}{2^k},\frac{j}{2^k}]} = \tilde{\eta}(x)$$

uniformly in x. The limit is a continuous function of compact support that satisfies the dilation relation

$$\tilde{\eta}(x) = \sum_l \alpha_l \tilde{\eta}(2x - l), \qquad \alpha_l = (\tilde{P}\delta^0)_l$$

and also $\tilde{\eta}(2^{-k}j) = (\tilde{P}^k\delta^0)_j$. We refer to [12] for a proof of these particular facts and to [13] for general statements concerning the limiting functions.

Let us consider the sequence

$$Z_j^m = \begin{cases} 0 & j \leq m \\ j & j \geq m. \end{cases}$$

The limiting process corresponding to $\tilde{P}^m Z^0$ is also convergent and we denote its limit by $\theta(x)$. Since

$$\delta^0 = Z^{-1} - 2Z^0 + Z^1$$

we get that

$$\tilde{\eta}(x) = \theta(x+1) - 2\theta(x) + \theta(x-1) \tag{162}$$

If the interpolatory procedure is at least first order, we must have a closed interval $I = [i_1, i_2]$ such that

$$\theta(x) = 0 \quad x \leq i_1 \qquad\qquad \theta(x) = x \quad x \geq i_2$$

Let us assume that $\theta(x)$ is a $\mathcal{C}^2$ function. We turn now to express the limiting process $P^k\delta^0$ for the reconstruction from hat-averages in terms of $\theta(x)$.

Notice that $\theta(2^{-k}j) = (\tilde{P}^k Z^0)_j$ (this is a consequence of the binary subdivision process

in the interpolatory framework). The relation

$$\delta^0 = Z^{-1} - 2Z^0 + Z^1 = Z^0_{j+1} - 2Z^0_j + Z^0_{j-1}$$

implies that $\{Z^0_j\}$ are the values of the second primitive of δ^0 on the grid $\{j\}^{+\infty}_{-\infty}$, thus (82) implies

$$(P\delta^0)_j = \frac{1}{2^{-2}}\left\{(\tilde{P}Z^0)_{j+1} - 2((\tilde{P}Z^0)_j + (\tilde{P}Z^0)_{j-1}\right\} \tag{163}$$

i.e., $\{(\tilde{P}Z^0)_j\}$ are the values of the second primitive of $P\delta^0$ on the grid $\{j/2\}^{+\infty}_{-\infty}$. It is easy to see then, that

$$\begin{aligned}(P^k\delta^0)_j &= \frac{1}{2^{-k}}\left\{(\tilde{P}^kZ^0)_{j+1} - 2((\tilde{P}^kZ^0)_j + (\tilde{P}^kZ^0)_{j-1}\right\} \\ &= \frac{1}{2^{-k}}\left\{\theta(2^{-k}(j+1)) - 2\theta(2^{-k}(j)) + \theta(2^{-k}(j-1))\right\}\end{aligned}$$

Since $\theta''(x)$ is a continuous function with compact support, we get that

$$\eta(x) = \lim_{m\to+\infty}\sum_{j=-\infty}^{+\infty}(P^m\delta^0)_j\chi_{[\frac{j-1}{2^k},\frac{j}{2^k}]} = \theta''(x).$$

uniformly in x. Notice that, (162) implies that

$$\eta(x) = \tilde{\eta}''(x+1) - 2\tilde{\eta}''(x) + \tilde{\eta}''(x-1)$$

and that supp $\eta \in I$.

The limit function satisfies the following relation (see [13])

$$\eta(x) = \sum_l \alpha_l\eta(2x-l), \qquad \alpha_l = (P\delta^0)_l.$$

In particular for the algorithms (114),(115), we have

$$\eta(x) = (2-\beta_1)\eta(2x) + \sum_{l=1}^{s}\beta_l\eta(2x-2l+1) - \frac{1}{2}(\beta_l+\beta_{l+1})\eta(2x-2l)$$

In Figures 27, 28, 29, 30, 31 and 32 we plot $\varphi^{0,8}_0$ and $\psi^{0,8}_0$ for $p = 2, 4, 6$ in the orthogonal and non-orthogonal cases of section 5. The circles denote the hat-averages of these functions in the intervals of X^8. Here $h_0 = 1/8$.

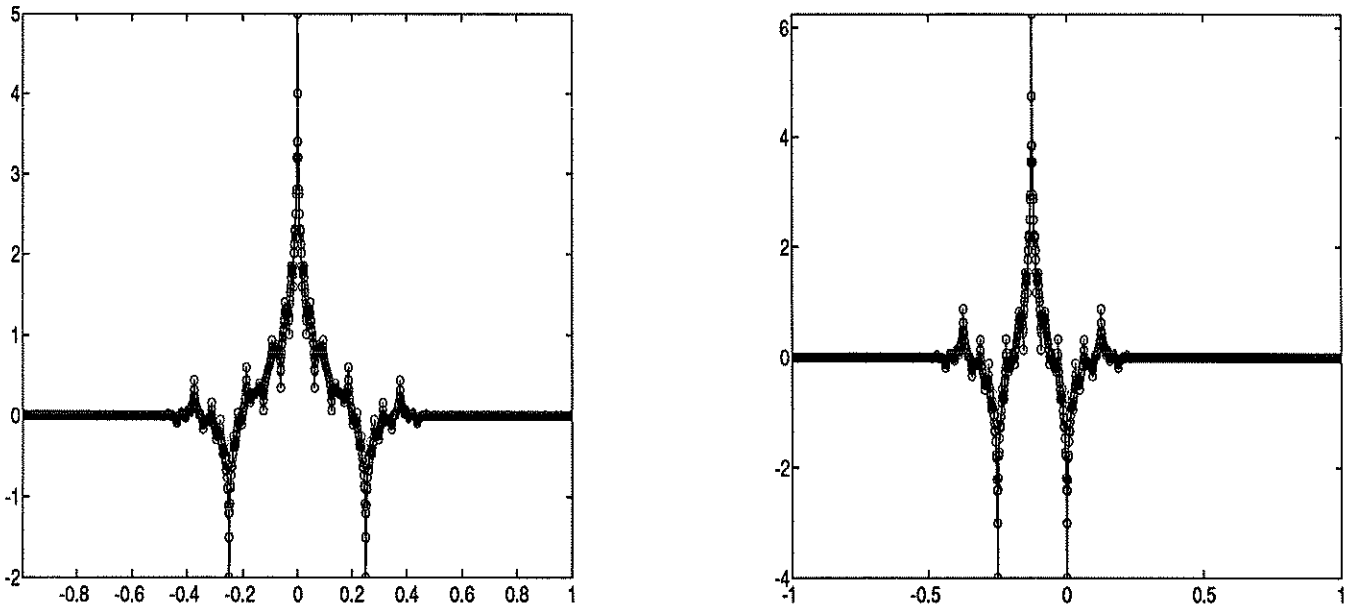

FIG. 27. *Left* $\varphi_0^{0,8}$, *Right* $\psi_0^{0,8}$, $p = 2$, *non-orthogonal*

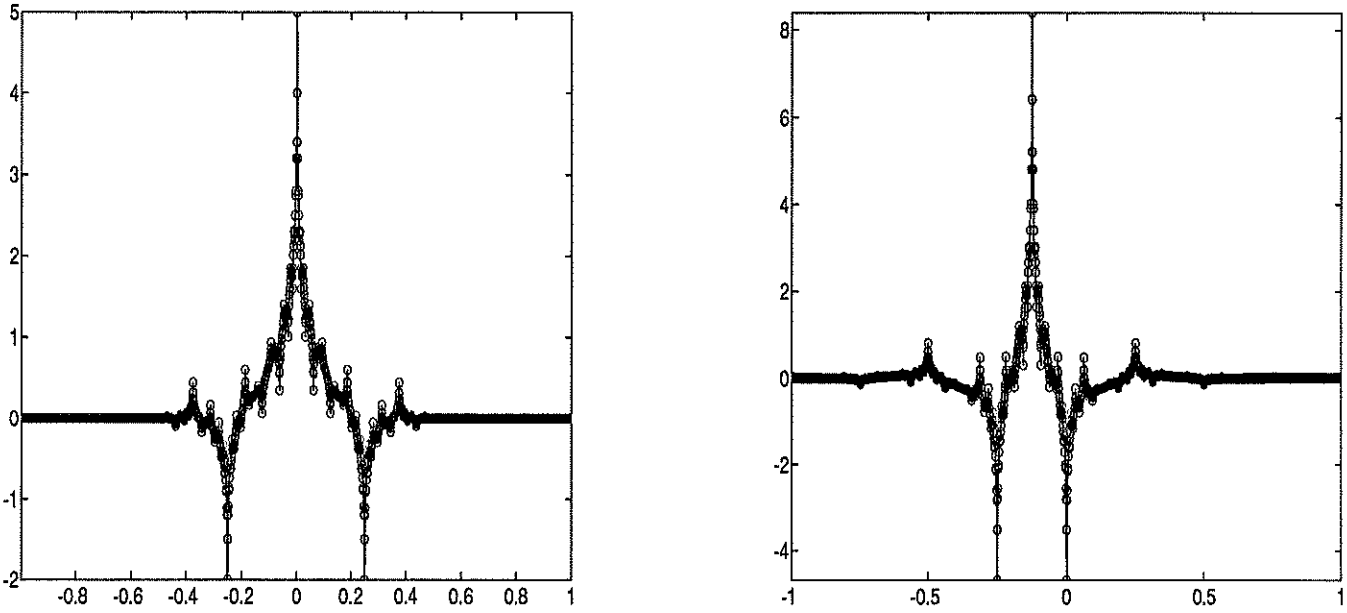

FIG. 28. *Left* $\varphi_0^{0,8}$, *Right* $\psi_0^{0,8}$, $p = 2$, *orthogonal*

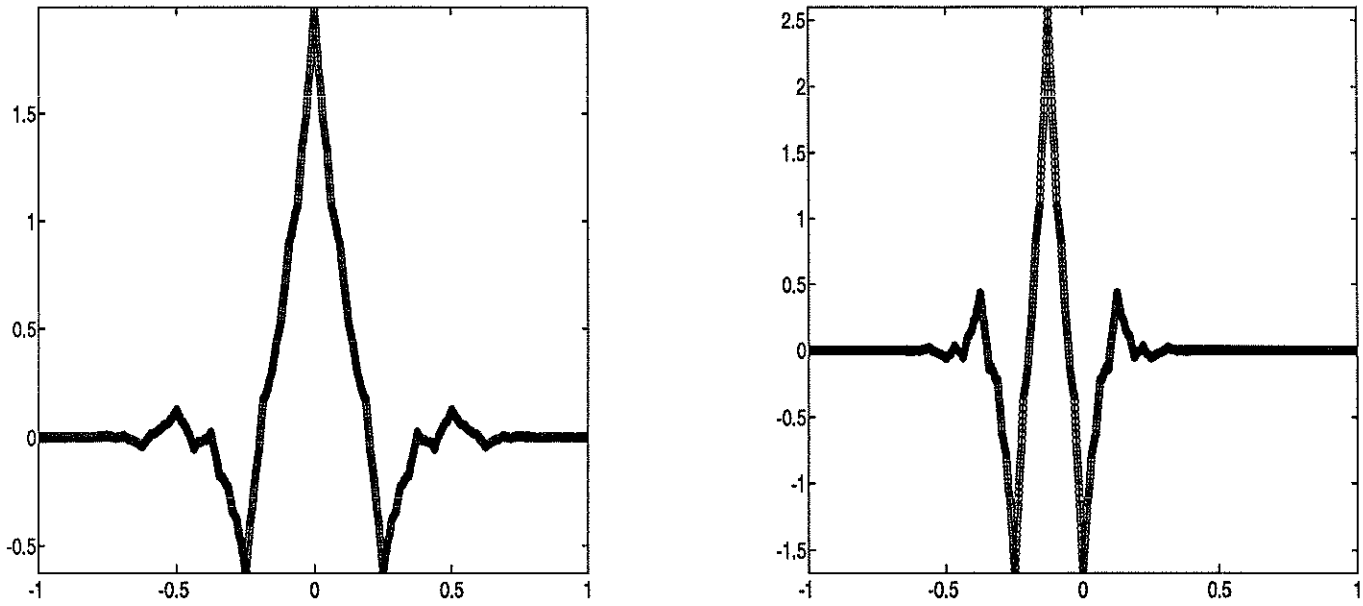

FIG. 29. *Left* $\varphi_0^{0,8}$, *Right* $\psi_0^{0,8}$, $p = 4$, *non-orthogonal*

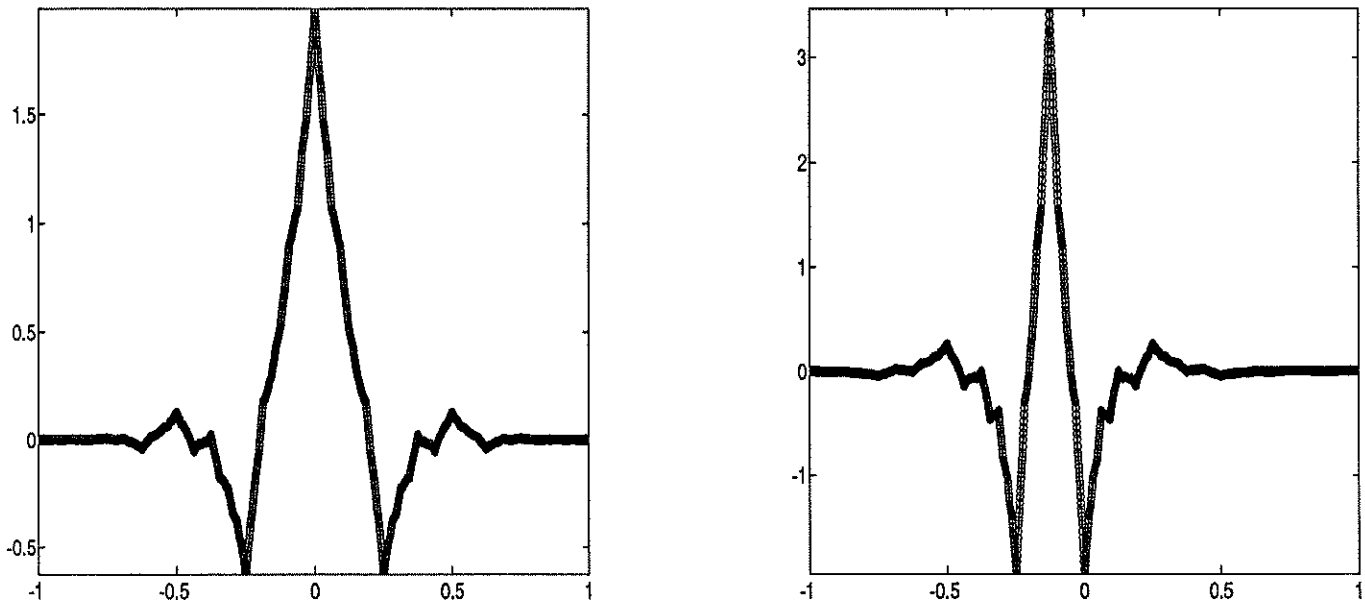

FIG. 30. *Left* $\varphi_0^{0,8}$, *Right* $\psi_0^{0,8}$, $p = 4$, *orthogonal*

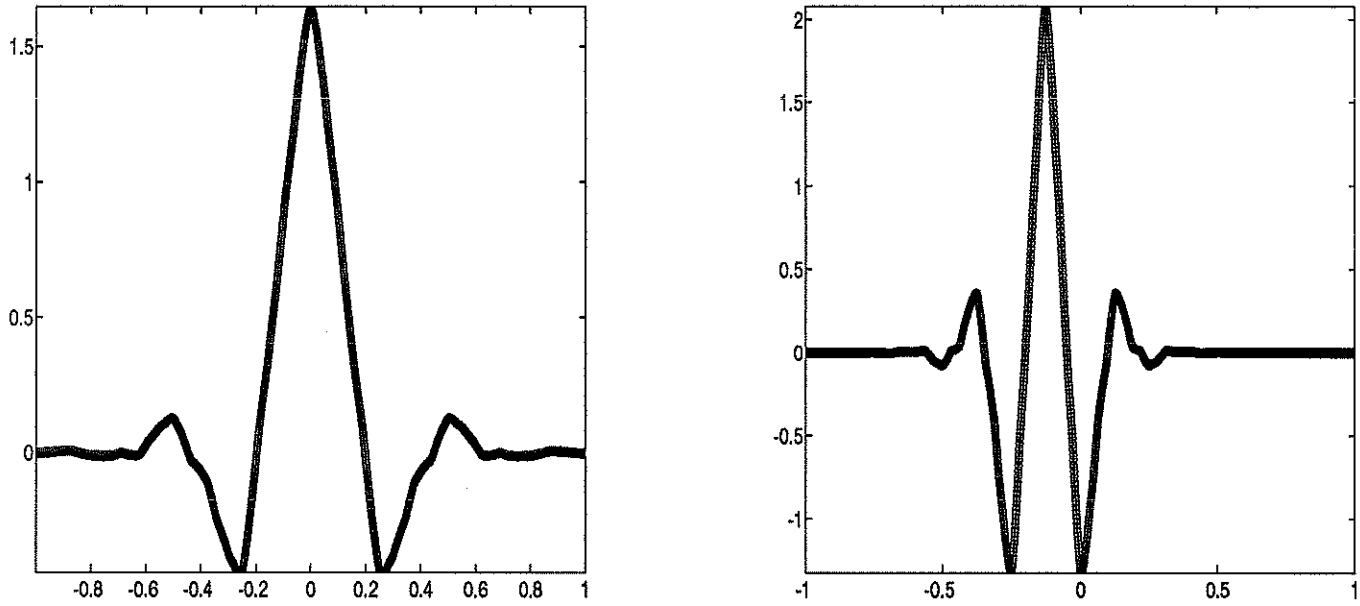

FIG. 31. *Left* $\varphi_0^{0,8}$, *Right* $\psi_0^{0,8}$, $p = 6$, *non-orthogonal*

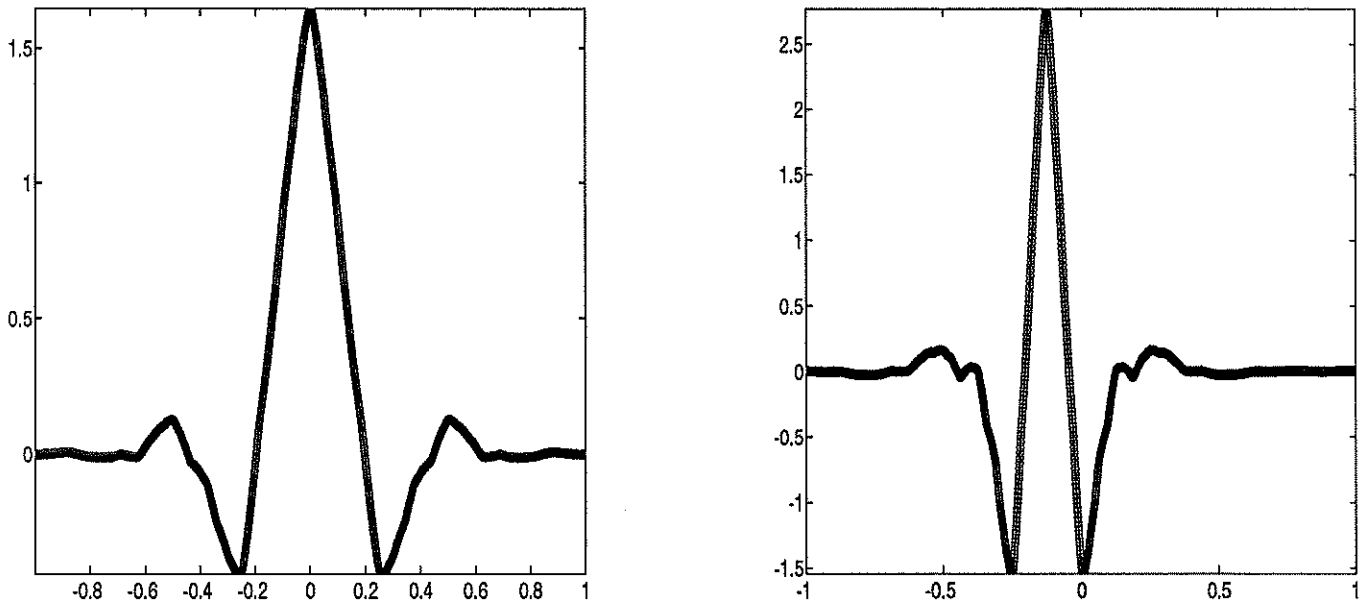

FIG. 32. *Left* $\varphi_0^{0,8}$, *Right* $\psi_0^{0,8}$, $p = 6$, *orthogonal*