DEPARTAMENTO DE MATEMÁTICA APLICADA

AVANCES EN LA MULTIRRSOLUCION DE HARTEN Y APLICACIONES

SERGIO AMAT PLATA

UNIVERSITAT DE VALENCIA
Servei de Publicacions
2002

Aquesta Tesi Doctoral va ser presentada a València el dia 19 de Octubre de 2001 , davant un Tribunal format per:

- Miguel Angel Henández Verón
- Jacques Liandrat
- David Javier López Medina
- Vicent Martínez García
- Rosa M Donat Beneito

Va ser dirigida per:
Prof. Dr. F. Arándiga I Llaudes, V Fº Candela Pomares

Depòsit legal:
I.S.B.N.: 84-370-556-3
Edita: Universitat de València
Servei de Publicacions
C/ Artes Gráficas, 13 bajo
46010 València
Spain
Telèfon: 963864115

UNIVERSITAT DE VALÈNCIA

AVANCES EN LA MULTIRRESOLUCIÓN DE HARTEN Y APLICACIONES ADVANCES IN MULTIRESOLUTION Á LA HARTEN AND APPLICATIONS

Sergio Amat Plata

Memoria presentada para optar al grado de Doctor en Matemáticas.

Departament de Matemàtica Aplicada
2001

Advances in Multiresolution á la Harten and applications
Avances en la Multirresolución de Harten y aplicaciones

Sergio Amat*

Multi-scale decomposition, discretization, nonlinear reconstruction, stability, conservation laws, local reconstructions.

*Departamento de Matemática Aplicada y Estadística. Universidad Politécnica de Cartagena (Spain). e-mail:sergio.amat@upct.es

Contents

List of Figures

List of Tables

1 Introducción

El análisis de Fourier proporciona una manera de representar funciones de cuadrado integrable en términos de sus componentes sinusoidales. La descomposición de Fourier es una herramienta básica para una gran variedad de aplicaciones en muchos campos de la ciencia. Sin embargo, tiene algún inconveniente, aquí destacaremos su carácter global. En Fourier, una singularidad aislada domina la conducta de todos los coeficientes de la descomposición y nos impide conseguir información precisa sobre la función lejos de la singularidad.

Descomposiciones "local-scale" proporcionan una mejor representación a este respecto. Típicamente, uno comienza con una sucesión finita, que es, de algún modo, asociada a la información discreta de una señal dada al nivel más fino de la resolución considerada. Procesando la señal en diferentes niveles de resolución, se puede escribir esta información discreta de una nueva manera. La nueva sucesión tiene el mismo cardinal que la primera (si se usa un esquema no redundante) y sus coeficientes representan por un lado los detalles a cada nivel de la resolución y por otro una aproximación "grosera" final a la señal original.

Uno de los aspectos fundamentales de este trabajo es profundizar en la Multirresolución introducida por A.Harten. Harten fue un reconocido matemático experto en teoría multigrid, soluciones de Ecuaciones en Derivadas Parciales y ondículas (wavelets).

En Multirresolución las funciones de $L^2(R)$ pueden ser computadas descomponiéndolas en una base ortonormal de wavelet. En teoría de wavelets, todas las operaciones se basan en una sucesión de transformaciones de base (ortogonal), por consiguiente la transformación inversa, es decir la recuperación de la señal discreta de partida, viene dada por las matrices adjuntas.

La base ortogonal wavelet está compuesta de dilataciones y translaciones de una sola función, el wavelet. El wavelet está íntimamente unido a la función scaling. Esta función satisface una relación de dilatación, la cual es de hecho, responsable de las propiedades de la descomposición.

En los métodos multigrid, la discretización por valores puntuales y la reconstrucción por interpolación son las dos herramientas esenciales que permiten la comunicación entre los diferentes niveles de resolución. Dada una sucesión de discretizaciones $\{\mathcal{D}_k\}$ y reconstrucciones $\{\mathcal{R}_k\}$, la manera más natural de transferir información de la celda k-ésima a la celda $k-1$-ésima más grosera es con el operador decimación $D_k^{k-1} = \mathcal{D}_{k-1}\mathcal{R}_k$. Similarmente, el operador predicción $P_{k-1}^k = \mathcal{D}_k\mathcal{R}_{k-1}$ es el candidato natural para transferir información de la celda $k-1$-ésima a la celda k-ésima, la próxima más fina. Harten vio que los operadores D_k^{k-1} y P_{k-1}^k pueden servir, respectivamente, como los utilizados en un esquema piramidal del tipo que se usan en procesamiento de imágenes.

Esta multirresolución puede verse como una generalización de la teoría wavelet, permitiéndonos trabajar en muy diversas situaciones y poder abordar gran cantidad de problemas. Además, esta multirresolución permite utilizar representaciones no lineales (datos-dependiente), lo que nos permitirá dar algoritmos más adaptados a los problemas a tratar. En este respecto, el operador de predicción utilizado en la teoría de Harten puede no ser lineal.

Por otra parte, las soluciones de leyes de conservación hiperbólicas desarrollan discontinuidades espontáneamente. Harten fue uno de los creadores de esquemas tipo ENO (esencialmente no oscilatorios), una clase de interpolación polinómica no lineal para obtener aproximaciones numéricas a los flujos de las leyes de conservación hiperbólicas sin generar oscilaciones.

A lo largo de esta tesis se profundizará en diversos aspectos de la multirresolución á la Harten y en algunas de sus aplicaciones. Así mismo, en la segunda parte, nos introduciremos en el mundo de las leyes de conservación desde un punto de vista numérico.

Comenzaremos con una breve exposición de la multirresolución introducida por Harten. La primera aplicación que estudiaremos será la detección, medida y clasificación de discontinuidades. Construiremos un algoritmo no lineal adaptado a señales perturbadas por ruido. El ruido introduce discontinuidades "ficticias" en la señal, lo que complica la detección de las verdaderas. El algoritmo se basará en el estudio de las diferencias divididas y como éstas se ven afectadas por la presencia de ruido.

En la sección cinco tratamos el concepto de estabilidad en la multirresolución. Nuestra noción de estabilidad proviene de la teoría de diferencias finitas y su papel es prevenir el crecimiento ilimitado del error provocado por perturbaciones iniciales. En el caso lineal, el problema está resuelto (ver [9], [31]); pero en el caso no lineal se deben modificar los algoritmos. En [10] los autores obtienen estabilidad para los algoritmos modificados en 1-D. Con estos algoritmos se demuestran resultados de estabilidad para cualquier reconstrucción, teniendo su importancia principal en el caso no lineal, ya que, las técnicas que hasta entonces existían sólo permiten demostrar la estabilidad en el caso lineal. Aquí, estudiaremos la estabilidad de los algoritmos no lineales en 2-D (ver [25], [26], [29] y [40] para el caso lineal).

La forma más natural de extender las ideas de 1-D es hacer producto tensorial. En [4] desarrollamos la estabilidad en 2-D para este caso, no obstante existe una gran cantidad de posibilidades diferentes.

En este trabajo introducimos un algoritmo modificado para cualquier tipo de reconstrucción, como un caso particular recuperamos el producto tensorial. Además las cotas serán obtenidas para diferentes tipos de normas.

En la sección diez presentamos nuestro actual trabajo. Se trata de un algoritmo no lineal para compresión de imágenes. La idea es conseguir una reconstrucción que utilice información de las zonas suaves de la imagen y que el esténcil no cruce las discontinuidades. La multirresolución tipo ENO es muy sensitiva al ruido y sus resultados no son óptimos como vimos en [5].

En la sección seis se presenta un esquema de "multirresolution-packets" no lineal. Los wavelets-packets fueron introducidos por R.R. Coifman [19], [20] y son una generalización de los wavelets que utilizan una descomposición basada en la elección de la base que minimice diversos criterios de entropía. Se estudian la estabilidad y las mejoras del nuevo algoritmo. La idea de este algoritmo es escoger una representación, en la que se localice espacio, frecuencia y suavidad adecuadamente.

Finalmente, estudiaremos leyes de conservación. Construiremos métodos locales adaptados a la presencia de "shocks" para leyes de consevación hiperbólicas. Nuestros métodos se han hecho con las mismas ideas introducidas por Marquina en [48], pero con una reconstrucción más simple (polinómica). Analizaremos el problema de los extremos locales. Introduciremos una extensión a quinto orden que compararemos con los métodos clásicos de alto orden. Por último, relacionaremos los conceptos de multirresolución y de leyes de conservación que hemos tratado en esta tesis.

2 Multirresolución á la Harten

Harten introduce su noción de análisis de multirresolución en [31] y posteriormente la completa en [32, 33] donde presenta los fundamentos teóricos para la representación multirresolutiva de datos.

Esta sección es una breve exposición de la construcción de un esquema de multirresolución *á la Harten.*

Comenzaremos introduciendo varias definiciones útiles.

Definición 1 *Un análisis de multirresolución es una sucesión de espacios lineales, $\{V^k\}$, que tienen bases numerables, que denotamos $\{\eta_i^k\}$, junto con una sucesión de operadores lineales $\{D_k^{k-1}\}$ de V^k a V^{k-1}, es decir*

$$D_k^{k-1} : V^k \to V^{k-1}, \qquad V^{k-1} = D_k^{k-1}(V^k).$$

El operador D_k^{k-1} se llama operador decimación.

Definición 2 *Decimos que P_{k-1}^k es un operador predicción para el esquema de multirresolución $(\{V^k\}, \{D_k^{k-1}\})$ si es un operador inverso de D_k^{k-1} en V^{k-1}, es decir*

$$P_{k-1}^k : V^{k-1} \to V^k, \qquad D_k^{k-1} P_{k-1}^k = I_{V^{k-1}}.$$

Notar que para P_{k-1}^k no se exige la linealidad.

Los espacios V^k representan los diferentes niveles de resolución. El operador $P_{k-1}^k D_k^{k-1} : V^k \to V^k$ produce aproximaciones para cada vector v^k en V^k de su información contenida al nivel $k-1$, es decir, $D_k^{k-1} v^k$. El error de la predicción

$$e^k := v^k - P_{k-1}^k v^{k-1} = (I_{V^k} - P_{k-1}^k D_k^{k-1}) v^k =: Q_k v^k \qquad (1)$$

es un vector en V^k tal que $D_k^{k-1}e^k = 0$, es decir, pertenece al núcleo asociado al operador decimación.

Consideramos una base en el núcleo $\mathcal{N}(D_k^{k-1})$

$$\mathcal{N}(D_k^{k-1}) = \text{span}\{\mu_j^k\}_j.$$

Sea $G_k : \mathcal{N}(D_k^{k-1}) \to \mathcal{G}^k$ el operador que asigna a cualquier $e^k \in \mathcal{N}(D_k^{k-1})$ la sucesión d^k de sus coordenadas en la base $\{\mu_j^k\}$ y sea E_k la inyección canónica $\mathcal{N}(D_k^{k-1}) \hookrightarrow V^k$.

Si consideramos $d^k := G_k Q_k v^k$, es fácil demostrar que existe una correspondencia biunívoca entre v^k y $\{d^k, v^{k-1}\}$.

Repitiendo este proceso para cada nivel deresolución, obtendremos un esquema de multirresolución formado por $(\{V^k\}_{k=0}^L, \{D_k^{k-1}\}_{k=1}^L)$ y una sucesión de operadores de predicción $\{P_{k-1}^k\}_{k=1}^L$ (lineales o no). Los algoritmos que computan esta transformación invertible, así como su inverso, son:

$v^L \quad \rightarrow \quad Mv^L$ (Encoding)

$$\left\{\begin{array}{l} \text{Do} \quad k = L, \ldots, 1 \\ v^{k-1} = D_k^{k-1} v^k \\ d^k = G_k(v^k - P_{k-1}^k v^{k-1}) \end{array}\right. \tag{2}$$

$Mv^L = \{v^0, d^1, \ldots, d^L\}$

$Mv^L \quad \rightarrow \quad M^{-1}Mv^L$ (Decoding)

$$\left\{\begin{array}{l} \text{Do} \quad k = 1, \ldots, L \\ v^k = P_{k-1}^k v^{k-1} + E_k d^k \end{array}\right. \tag{3}$$

Nos referiremos a Mv^L como la representación multirresolutiva de v^L, y a los algoritmos (2) y (3) como las transformaciones de multirresolución directa e inversa, respectivamente.

Veamos como una sucesión de operadores decimación puede construirse a partir de una *sucesión anidada de operadores discretización.*

Definición 3 *Sea $\mathcal{D}$ un operador lineal en un espacio lineal $\mathcal{F}$, con rango V. Si V tiene una base numerable, $\{\eta_i\}$, decimos que $\mathcal{D}$ es un operador discretización de $\mathcal{F}$ y, para cada $f \in \mathcal{F}$, consideramos a $v = \mathcal{D}f$ como la discretización de f al nivel de la resolución especificado por V.*

$$\mathcal{D} : \mathcal{F} \to V, \qquad \textit{donde} \quad V = \mathcal{D}(\mathcal{F}) = span\{\eta_i\}.$$

Definición 4 *Sea $\{\mathcal{D}_k\}$ una sucesión de operadores de discretización en $\mathcal{F}$*

$$\mathcal{D}_k : \mathcal{F} \to V^k, \qquad \mathcal{D}_k(\mathcal{F}) = V^k = span\{\eta_i^k\}.$$

Decimos que la sucesión $\{\mathcal{D}_k\}$ es anidada si para todo k y toda $f \in \mathcal{F}$

$$\mathcal{D}_k f = 0 \Rightarrow \mathcal{D}_{k-1} f = 0. \tag{4}$$

La propiedad de sucesión anidada implica que la información discreta en cada uno de los niveles de resolución es idéntica.

Una sucesión anidada de discretizaciones define una sucesión de operadores decimación, y así, un esquema de multirresolución. Este resultado se sigue del siguiente lema:

Lema 1 *Si $\{\mathcal{D}_k\}$ es una sucesión anidada de discretización, entonces la aplicación de V^k a V^{k-1} definida como sigue:*

Para cada $v \in V^k$ tomar un $f \in \mathcal{F}$ tal que $v = \mathcal{D}_k f$ y asignar $u := \mathcal{D}_{k-1} f$, *está bien definida.*

Cada operador decimación se define entonces como sigue: Para cualquier $v^k \in V^k$, sea $f \in \mathcal{F}$ tal que $\mathcal{D}_k f = v^k$; entonces $D_k^{k-1} v^k = \mathcal{D}_{k-1} f$. El lema 1 implica que la definición es independiente de f. Así tenemos

$$D_k^{k-1} \mathcal{D}_k = \mathcal{D}_{k-1} \tag{5}$$

(D_k^{k-1} es un operador lineal). Además, para una sucesión anidada de discretización, (5) define un operador que aplica V^k a V^{k-1}. Para ver esto, sea $u \in V^{k-1}$ y tómese $f \in \mathcal{F}$ tal que $u = \mathcal{D}_{k-1} f$, y sea $v = \mathcal{D}_k f$. Claramente $v \in V^k$ y (5) implica

$$D_k^{k-1} v = D_k^{k-1} (\mathcal{D}_k f) = \mathcal{D}_{k-1} f = u.$$

Dado $v \in V^k$, una $f \in \mathcal{F}$ verificando $v = \mathcal{D}_k f$ se llama una reconstrucción de v en $\mathcal{F}$.

Los operadores de predicción se construyen usando una sucesión de operadores de reconstrucción apropiados.

Definición 5 *Decimos que $\mathcal{R}$*

$$\mathcal{R} : V \to \mathcal{F}, \qquad V = \mathcal{D}(\mathcal{F})$$

es un operador de reconstrucción para $\mathcal{D}$ en V si es un operador inverso por la derecha de $\mathcal{D}$, es decir $\mathcal{D}\mathcal{R} = I_V$. Notemos que para $\mathcal{R}$ no se exige la linealidad.

Dada una sucesión de operadores de discretización $\{\mathcal{D}_k\}$ y una de correspondientes operadores de reconstrucción $\{\mathcal{R}_k\}$, un operador inverso por la derecha de D_k^{k-1} puede definirse ahora fácilmente como sigue

$$P_{k-1}^k := \mathcal{D}_k \mathcal{R}_{k-1} : V^{k-1} \to V^k. \tag{6}$$

El operador de predicción definido es un inverso por la derecha de D_k^{k-1} .

$$\begin{tikzcd}
& \mathcal{F} \arrow[dr, "\mathcal{D}_{k-1}"] & \\
V^k \arrow[ur, "\mathcal{R}_k"] \arrow[rr, bend left=90, "D_k^{k-1}"] & & V^{k-1} \arrow[dl, "\mathcal{R}_{k-1}"] \arrow[ll, bend left=90, "P_{k-1}^k"] \\
& \mathcal{F} \arrow[ul, "\mathcal{D}_k"] &
\end{tikzcd}$$

Fig. 1: Definición de operadores

Ahora, presentaremos tres casos particulares donde la multirresolución está basada: en discretización por valores puntuales, por promedios en celdas y por promedios basados en la función hat respectivamente.

2.1 Multirresolución puntual

Consideramos un conjunto de mallas anidadas:

$$X^k = \{x_j^k\}_{j=0}^{J_k}, \quad x_j^k = jh_k, \quad h_k = 2^{-k}/J_0, \quad J_k = 2^k J_0,$$

donde J_0 es un número entero.

Definimos

$$\mathcal{D}_k : \mathcal{C}([0,1]) \longrightarrow V^k \qquad \bar{f}_j^k = (\mathcal{D}_k f)_j = f(x_j^k), \quad 0 \leq j \leq J_k. \tag{7}$$

donde $\mathcal{C}([0,1])$ es el espacio de las funciones continuas en $[0,1]$ y V^k el espacio de sucesiones de dimensión $J_k + 1$.

Como $\bar{f}_j^{k-1} = f(x_j^{k-1}) = f(x_{2j}^k) = \bar{f}_{2j}^k$, obtenemos los elementos de un espacio V^{k-1} de los elementos del V^k más fino tomando sólo los elementos con índice par. Por lo tanto $(D_k^{k-1})_{ij} = \delta_{2i,j}$ y así $(G_k)_{ij} = \delta_{2i-1,j}$ y $(E_k)_{ij} = \delta_{i,2j-1}$.

Sea $\mathcal{I}_{k-1}(x;\bar{f}^{k-1})$ una función interpoladora tal que $\mathcal{I}_{k-1}(x_j^{k-1};\bar{f}^{k-1}) = \bar{f}_j^{k-1}$. Con esta función, podemos obtener una aproximación a partir de los valores de la malla $k-1$-ésima a $\bar{f}_j^k$, es decir,

$$\tilde{f}_j^k = \mathcal{I}_{k-1}(x_j^k, \bar{f}^{k-1}). \qquad . \quad (8)$$

Y así $(P_{k-1}^k \bar{f}^{k-1})_j = \mathcal{I}_{k-1}(x_j^k;\bar{f}^{k-1})$.

Por otra parte como $\bar{f}_{2j}^k = \tilde{f}_{2j}^k$, *los coeficientes "scaling"* $\{d_j\}_{j=0}^{J_{k-1}}$ serán los errores interpolatorios que obtenemos al predecir los valores de índice impar de un nivel V^k de los elementos de V^{k-1}:

$$d_j = \bar{f}_{2j-1}^k - \tilde{f}_{2j-1}^k = \bar{f}_{2j-1}^k - \mathcal{I}_{k-1}(x_{2j-1}^k, \bar{f}^{k-1}) \qquad 1 \leq j \leq J_{k-1}.$$

Notar que los conjuntos $\bar{f}^k$y $\{d, \bar{f}^{k-1}\}$ tienen el mismo número de elementos.

La descomposición multiescala de los datos originales $\bar{f}^L$ es

$$M\bar{f}^L = \{\bar{f}^0, d^1, \ldots, d^L\}$$

y puede obtenerse usando el algoritmo:

$$\bar{f}^L \to M\bar{f}^L \begin{cases} \text{Do} \quad k = L, \ldots, 1 & \\ \quad \bar{f}_j^{k-1} = \bar{f}_{2j}^k & 0 \leq j \leq J_{k-1} \\ d_j^k \quad = \bar{f}_{2j-1}^k - \mathcal{I}(x_{2j-1}^k;\bar{f}^{k-1}) & 1 \leq j \leq J_{k-1} \end{cases} \qquad (9)$$

Para recuperar los datos originales utilizaremos el algoritmo:

$$M\bar{f}^L \to M^{-1}M\bar{f}^L \begin{cases} \text{Do} \quad k = 1, \ldots, L & \\ \bar{f}_{2j-1}^k = \mathcal{I}(x_{2j-1}^k;\bar{f}^{k-1}) + d_j^k & 1 \leq i \leq J_{k-1} \\ \quad \bar{f}_{2j}^k = \bar{f}_j^{k-1} & 0 \leq j \leq J_{k-1} \end{cases} \qquad (10)$$

2.2 Multirresolución por promedios en celda

Consideramos el mismo conjunto de mallas anidadas que en la sección 2.1. La discretización por promedios en celda se define como sigue:

$$\mathcal{D}_k : L^1[0,1] \longrightarrow V^k, \qquad \bar{f}_j^k = (\mathcal{D}_k f)_j = \frac{1}{h_k}\int_{x_{j-1}^k}^{x_j^k} f(x)dx, \quad 1 \leq j \leq J_k, \tag{11}$$

donde $L^1[0,1]$ es el espacio de las funciones absolutamente integrables en $[0,1]$ y V^k será el espacio de sucesiones con J_k componentes.

Dada la relación

$$\bar{f}_j^{k-1} = \frac{1}{h_{k-1}}\int_{x_{j-1}^{k-1}}^{x_j^{k-1}} f(x)dx = \frac{1}{2h_k}\int_{x_{2j-2}^k}^{x_{2j}^k} f(x)dx = \frac{1}{2}(\bar{f}_{2j-1}^k + \bar{f}_{2j}^k),$$

se deduce que $\{\bar{f}_j^k\}_{j=1}^{J_k}$, $k = L-1, \ldots, 1$, puede evaluarse directamente de $\{\bar{f}_j^L\}_{j=1}^{J_L}$ sin un conocimiento explícito de la función original $f(x)$. Además, los errores de predicción verificarán

$$0 = \frac{(e_{2j-1}^k + e_{2j}^k)}{2}$$

y los operadores G_k y E_k pueden ser definidos como sigue

$$d_j^k = e_{2j-1}^k \qquad 1 \leq j \leq J_{k-1} \tag{12}$$

$$e_{2j-1}^k = d_j^k \qquad 1 \leq j \leq J_{k-1} \tag{13}$$

$$e_{2j}^k = -d_j^k \qquad 1 \leq j \leq J_{k-1} \tag{14}$$

Definimos, ahora, la sucesión $\{F_j^k\}$ en la celda k-ésima como

$$F_j^k = h_k \sum_{i=1}^{j} \bar{f}_i^k = \int_0^{x_j^k} f(x)dx = F(x_j^k) \quad \Rightarrow \quad \bar{f}_j^k = \frac{F_j^k - F_{j-1}^k}{h_k}. \tag{15}$$

Si $F(x)$ es la primitiva de $f(x)$ entonces la sucesión $\{F_j^k\}$ corresponde a una discretización puntual de $F(x)$ en la celda k-ésima. Conociendo $\{\bar{f}_j^k\}_{j=1}^{J_k}$ podemos evaluar $\{F_j^k\}_{j=1}^{J_k}$ (y vice-versa) usando (15).

Ahora, denotamos por $\mathcal{I}_{k-1}(x; F^{k-1})$ una reconstrucción tal que

$$\mathcal{I}_{k-1}(x_j^{k-1}; F^{k-1}) = F_j^{k-1}.$$

Así, podemos obtener una aproximación, $\tilde{f}_j^k$, a $\bar{f}_j^k$ usando (15):

$$\tilde{f}_j^k = (\tilde{F}_j^k - \tilde{F}_{j-1}^k)/h_k = (\mathcal{I}_{k-1}(x_j^k, F^{k-1}) - \mathcal{I}_{k-1}(x_{j-1}^k, F^{k-1}))/h_k. \qquad (16)$$

De $F_{2j}^k = F(x_{2j}^k) = F(x_j^{k-1}) = F_j^{k-1}$, se obtiene

$$\tilde{f}_{2j-1}^k = (\tilde{F}_{2j-1}^k - F_{j-1}^{k-1})/h_k \quad \text{y} \quad \tilde{f}_{2j}^k = (F_j^{k-1} - \tilde{F}_{2j-1}^k)/h_k.$$

Consideramos $d^k = \{d_j^k\}_{j=1}^{J_k}$ los errores de la predicción obtenidos cuando aproximamos $\{\bar{f}_{2j-1}^k\}$ a partir de $\bar{f}^{k-1}$. Como $\bar{f}_{2j-1}^k - \tilde{f}_{2j-1}^k = -(\bar{f}_{2j}^k - \tilde{f}_{2j}^k)$, deducimos que d^k contiene toda la información de los errores de la predicción.

La descomposición multiescala de $\bar{f}^L$ es $M\bar{f}^L = \{\bar{f}^0, d^1, \ldots, d^L\}$. Los algoritmos directo e inverso son ahora:

$$\bar{f}^L \to M\bar{f}^L \left\{ \begin{array}{ll} \text{Do} \quad k = L, \ldots, 1 & \\ \quad \bar{f}_j^{k-1} = \frac{1}{2}(\bar{f}_{2j-1}^k + \bar{f}_{2j}^k) & 1 \le j \le J_{k-1} \\ \quad d_j^k = \bar{f}_{2j-1}^k - (\mathcal{I}(x_{2j-1}^k; F^{k-1}) - F_{j-1}^{k-1})/h_k & 1 \le j \le J_{k-1} \end{array} \right. \qquad (17)$$

$$M\bar{f}^L \to M^{-1}M\bar{f}^L \left\{ \begin{array}{ll} \text{Do} \quad k = 1, \ldots, L & \\ \quad \bar{f}_{2j-1}^k = (\mathcal{I}(x_{2j-1}^k; F^{k-1}) - F_{j-1}^{k-1})/h_k + d_j^k & 1 \le j \le J_{k-1} \\ \quad \bar{f}_{2j}^k = 2\bar{f}_j^{k-1} - \bar{f}_{2j-1}^k & 1 \le j \le J_{k-1} \end{array} \right. \qquad (18)$$

Remark 2.1 *Es posible obtener algoritmos directos sin pasar por la función primitiva (ver los trabajos de Harten).*

2.3 Multirresolución Hat-average

Nuevamente, consideramos el mismo conjunto de mallas anidadas que en la sección 2.1.

La discretización estará basada en la función hat:

$$w(x) = \begin{cases} 1+x & -1 \leq x \leq 0 \\ 1-x & 0 < x \leq 1 \\ 0 & \textit{en otro caso} \end{cases}$$

Definimos

$$\bar{f}_j^k = (\mathcal{D}_k f)_i = \int f(x) w_j^k(x) dx, \quad \textit{donde } w_j^k(x) = \frac{1}{h_k} w(\frac{x}{h_k} - j) \;\; 1 \leq j \leq J_k. \quad (19)$$

La descomposición multiescala de los datos originales $\bar{f}^L$ es

$$M\bar{f}^L = \{\bar{f}^0, d^1, \ldots, d^L\}$$

y puede obtenerse usando el algoritmo:

$$\bar{f}^L \to M\bar{f}^L \begin{cases} \text{Do} \quad k = L, \ldots, 1 \\ \quad \bar{f}_j^{k-1} = \frac{1}{4}(\bar{f}_{2j-1}^k + 2\bar{f}_{2j}^k + \bar{f}_{2j+1}^k) & 1 \leq j \leq J_{k-1} - 1 \\ d_j^k \quad = \bar{f}_{2j-1}^k - (P_{k-1}^k \bar{f}^{k-1})_{2j-1} & 1 \leq j \leq J_{k-1} \end{cases} \cdot \quad (20)$$

Para recuperar los datos originales utilizaremos el algoritmo:

$$M\bar{f}^L \to M^{-1}M\bar{f}^L \begin{cases} \text{Do} \quad k = 1, \ldots, L \\ \bar{f}_{2j-1}^k = (P_{k-1}^k \bar{f}^{k-1})_{2j-1} + d_j^k & 1 \leq j \leq J_{k-1} \\ \quad \bar{f}_{2j}^k = 2\bar{f}_j^{k-1} - \frac{1}{2}(\bar{f}_{2j-1}^k + \bar{f}_{2j+1}^k) & 1 \leq j \leq J_{k-1} - 1 \end{cases} \quad (21)$$

donde $(P_k^k \hat{f}^{k-1})_{2j-1} = \frac{1}{{h_k}^2}(q_j^{k-1}(x_{2j-2}^k) - 2q_j^{k-1}(x_{2j-1}^k) + q_j^{k-1}(x_{2j}^k))$ y q_j^{k-1} es una función interpoladora en $[x_{j-1}^k, x_j^k]$ de la segunda primitiva de f.

REMARK 2.2 *Todos los detalles de este tipo de reconstrucción así como algoritmos directos sin necesidad de pasar por la segunda primitiva se pueden encontrar en [9] y [10].*

3 Reconstrucción no lineal: ENO

Las reconstrucciones más usadas corresponden a interpolaciones polinómicas a trozos [31, 32, 33] y [9]. Si se usa una interpolación (lineal) de alto orden su estencil cruzará en más ocasiones las singularidades existentes, disminuyendo en esos casos su eficacia. En [39], Harten et al. introducen una interpolación polinómica a trozos dependiente de los datos, y por lo tanto no lineal, la interpolación·ENO (esencialmente no oscilatoria). La idea básica de esta interpolación es agrandar la región de alta resolución construyendo interpoladores polinómicos que usen sólo información de regiones suaves de la función.

Acontinuación recordaremos brevemente esta interpolación.

3.1 Interpolación ENO

Con el objetivo de hacer la presentación más simple consideraremos una malla uniforme $X = \{x_j\}$ en $[0, 1]$ y sea $h = x_{j+1} - x_j$ (en [1, 34] se generaliza a varias dimensiones y a mallas no uniformes).

Sea $H(x) \in \mathcal{C}[0, 1]$ y $\mathcal{D}H = (H_j)_j$, donde $H_j = H(x_j)$. Sea $\mathcal{I}(x; \mathcal{D}H)$ un interpolador polinómico de $H(x)$ en la malla X.

Si consideramos una interpolación de orden r, tendremos

$$\mathcal{I}(x; \mathcal{D}H) = q_j(x; \mathcal{D}H) \qquad \text{para} \quad x \in [x_{j-1}, x_j],$$

donde $q_j(x; \mathcal{D}H)$ es un polinomio de grado $r - 1$ tal que $q_j(x_{j-1}; \mathcal{D}H) = H_{j-1}$ y $q_j(x_j; \mathcal{D}H) = H_j$. El conjunto de r nodos asociados al polinomio $q_j(x; \mathcal{D}H)$ forma el *estencil*, $\mathcal{S}_j$, asociado al intervalo $[x_{j-1}, x_j]$. Los nodos x_{j-1} y x_j deben estar en $\mathcal{S}_j$.

La idea de la interpolación ENO es construir un estencil $\mathcal{S}_j$ utilizando información sólo de regiones suaves de $H(x)$.

Para cada intervalo $[x_{j-1}, x_j]$, consideramos todos los posibles estencils de cardinal $r \geq 2$ y que incluyan a los nodos x_{j-1} y x_j,

$$\{x_{j-r+1}, \ldots, x_j\}, \cdots, \{x_{j-1}, \ldots, x_{j+r-2}\}$$

Se trata de seleccionar de entre todos los candidatos aquel que cumpla nuestros requerimientos. Sea $i(j)$, el índice correspondiente al segundo punto del estencil seleccionado. Notar que si $r = 2$, $\mathcal{S}_j = \{x_{j-1}, x_j\}$ y no es necesaria ninguna selección, supongamos entonces que $r > 2$.

En [39], los autores describen dos procedimientos de selección de los estencils:

Algoritmo I. Elección Jerárquica:

```
Tomar $i_0(j) = j$
$for \quad l = 0, \ldots, r-3$
  $if \quad |H(x_{i_l(j)-2}, \ldots, x_{i_l(j)+l})| < |H(x_{i_l(j)-1}, \ldots, x_{i_l(j)+l+1})|$
    $i_{l+1}(j) = i_l(j) - 1$
  $else$
    $i_{l+1}(j) = i_l(j)$
  $end$
$end$
$i(j) = i_{r-2}(j)$.
```

Algoritmo II. Elección no Jerárquica:

```
Tomar $i(j)$ tal que
$|H(x_{i(j)-1}, \ldots, x_{i(j)+r-2})| = \min\{|H(x_{l-1}, \ldots, x_{l+r-2})|, \quad j-r+2 \leq l \leq j\}$.
```

Notar que si $j - r + 2 \leq i(j) \leq j$, entonces x_{j-1}, $x_j \in \mathcal{S}_j$, en ambos casos. Mediante $H(*, \ldots, *)$ se han denotado las diferencias divididas de la función H.

En general usaremos el Algoritmo I (menos costoso) salvo cuando estemos interesados en singularidades débiles, ya que, en este caso pueden haber problemas con este esquema (ver [27]).

4 Detection, measurement and classification of discontinuities

The problem of detection, classification and measurement of discontinuities appears in many applications in science and technology. Some of these processes produce piecewise smooth data, that is functions with a small number of discontinuities compared to the number of sampled data. Assume that the input function is corrupted by an additive random noise $\hat{f} = f + n$, we would like to find the discontinuities of the signal f. The noise disturbs the data thus the problem is complex. It is difficult to distinguish the true discontinuities from the function and the false discontinuities from the noise. The noise added to the signal appears as small oscillatory deviations from the curve. A non-linear detector algorithm is presented. The main advantage of our algorithm is that we can consider noise larger than the classic methods.

A function has a discontinuity of degree k at a point, if the k th-order left and right derivatives at that point are different. Discontinuities are classified by their degrees and measured by their sizes, that is, the difference of the derivatives.

In [10], divided differences were studied to obtain the possible discontinuities. Our detection algorithm is based on this study and on the subcell resolution technique introduced by Harten. However, when we consider signals corrupted by noise, we have to modify the detecting mechanisms, since the algorithm should be adapted to the introduced noise. Our new algorithm will detect true singularities only and not singularities introduced by the noise. In the examples we will see that it is possible to consider noise of very large size.

Now we will recall the subcell resolution technique.

4.1 The Subcell Resolution Technique.

Let us assume that $H(x)$ is a continuous function with a corner at $x_d \in (x_{j-1}, x_j)$. Then, the ENO interpolants satisfy

$$H(x) = q_{j-1}(x) + O(h^r), \quad x \in [x_{j-2}, x_{j-1}] \tag{22}$$

$$H(x) = q_{j+1}(x) + O(h^r), \quad x \in [x_j, x_{j+1}] \tag{23}$$

The location of the corner, x_d, can be recovered using the following function:

$$G_j(x) = q_{j+1}(x) - q_{j-1}(x) \tag{24}$$

Using Taylor expansion in regions of smoothness, it is not hard to prove that

$$G_j(x_{j-1}) \times G_j(x_j) = a(a-1)[H']^2_{x_d} h^2 + O(h^3)$$

where $x_d = x_j - ah$, $0 < a < 1$ and $[H']_{x_d}$ denotes the jump of the derivative at x_d.

Therefore, if h is sufficiently small, there is a root of G_j in (x_{j-1}, x_j) be such that $G_j(\theta_j) = 0$. In general, it can be proven [27] that

$$|\theta_j - x_d| = O(h^r)$$

REMARK 4.1 *If the function is a piecewise polynomial with a corner in x_d then $x_d = \theta_j$.*

Working with cell-average and hat-average, via first and second primitive (see [9] and [10]), we can detect jumps and delta singularities. With these multiresolutions we can detect "weaker" singularities also, that is to say, with the cell-average we can detect corners and with the hat-average we can detect corners and jumps.

In these cases it becomes very important to isolate cells that are suspected of harboring a singularity.

On the other hand, we know that when $H(x)$ has a discontinuity in its $m+1$st derivative at $x_d \in (x_{j-1}, x_j)$, it can be approximated (for sufficiently small h) by the unique root of $G_j^{(m)}(z) = q_{j+1}^{(m)}(z) - q_{j-1}^{(m)}(z) = 0$. Thus, if (x_{j-1}, x_j) is suspected of containing a singularity (stencil selection, see [10]), we check whether

$$G_j^{(m)}(x_{j-1}) \cdot G_j^{(m)}(x_j) < 0. \tag{25}$$

If this is the case, we conclude that there is a root of $G_j^{(m)}(z)$ in (x_{j-1}, x_j).

A careful analysis of the functions $G_j^{(m)}(x)$ for $m = 0, 1, 2$ can help to determine whether or not a singularity lies at a *suspicious* grid point.

Since we are interesting in jumps and corners we will consider the cell-average framework.

4.2 Full detection mechanism

As we said before, we will work with signals perturbed with noise for which we will assume some conditions. For our algorithm we will need to know some bound ϵ of the introduced noise. The knowing of noise bounds is not a big restriction. In the classic detectors, it is supposed that the noise is modeled by a gaussian of which we know the mean μ and the variance σ. With this information we can find bounds since $Pr(f \in [\mu - 2\sigma, \mu + 2\sigma]) = 0.95$.

When we don't have any information of the noise (for example picture from an airplane with a lot of fog), we can generate a decreasing succession of parameters ϵ_k in order to obtain the possible discontinuities. We keep the discontinuities obtained in a such ϵ_{k_0} if the next parameter ϵ_{k_0+1} knowledge the detection mechanism getting a number limitless of discontinuities.

	$x_d \in (x_{j-1}, x_j)$		$x_d = x_j$	
z	$G_j(z)$	$G'_j(z)$	$G_j(z)$	$G'_j(z)$
x_{j-1}	$(a-1)h[H']_{x_d}$	$[H']_{x_d}$	$-h[H']_{x_d}$	$[H']_{x_d}$
x_j	$ah[H']_{x_d}$	$[H']_{x_d}$	$O(h^{p+2})$	$[H']_{x_d}$
x_{j+1}	$(a+1)h[H']_{x_d}$	$[H']_{x_d}$	$h[H']_{x_d}$	$[H']_{x_d}$

Table 1: Jump in $f(x)$ ($[H']_{x_d} \neq 0$) .

	$x_d \in (x_{j-1}, x_j)$		$x_d = x_j$	
z	$G'_j(z)$	$G''_j(z)$	$G'_j(z)$	$G''_j(z)$
x_{j-1}	$(a-1)h[H'']_{x_d}$	$[H'']_{x_d}$	$-h[H'']_{x_d}$	$[H'']_{x_d}$
x_j	$ah[H'']_{x_d}$	$[H'']_{x_d}$	$O(h^{p+1})$	$[H'']_{x_d}$
x_{j+1}	$(a+1)h[H'']_{x_d}$	$[H'']_{x_d}$	$h[H'']_{x_d}$	$[H'']_{x_d}$

Table 2: Corner in $f(x)$ ($[H'']_{x_d} \neq 0$).

Tables 1 and 2 (which are constructed via Taylor expansions) reflect the behavior of the functions $G^{(m)}$ near the different types of singularities (see [10] for more details). When we are considering signals perturbed with noise, the problem is more difficult. The idea is to study how this noise affects the divided differences. Notice you for example that if $||f_i - \hat{f}_i|| < \epsilon$ then for the divided differences of order 4 we have $||H[i;4] - \hat{H}[i;4]|| < \frac{8}{6}\frac{\epsilon}{h^3}$.

Our strategy to detect singularities is based on tables 1, 2 and on the presence of noise. It is summarized in Table (3).

REMARK 4.2 *We only need a vector of data not the complete function.*

```
$ave := (|G'(x_{j-1})| + |G'(x_j)| + |G'(x_{j+1})|)/3$
$nn = 0$
if  ( $A$ singularity can exist at $x_j$, adapted stencil selection)
      if  $(G_j'(x_{j-1})G_j'(x_{j+1}) \leq -64(\frac{\epsilon}{h})^2$ and $|G_j'(x_j)| + 8\epsilon \leq h\min(|G_j'(x_{j-1})|, |G_j'(x_{j+1})|)$
      and $((8\frac{\epsilon}{h^2} \leq \min(G_j''(x_{j-1}), G_j''(x_j))$ or $\max(G_j''(x_{j+1}), G_j''(x_j))) \leq -8\frac{\epsilon}{h^2}))$
      and $|G_j'(x_{j+1})| + 8\frac{\epsilon}{h} + 8\frac{\epsilon}{h^2} < |G_j''(x_{j+1})|$
      and $|G_j'(x_{j-1})| + 8\frac{\epsilon}{h} + 8\frac{\epsilon}{h^2} < |G_j''(x_{j-1})|)$  then
            there is a corner at $x_j$
            $nn = 1$
      elseif  $(G_j(x_{j+1})G_j(x_{j-1}) \leq -16\epsilon^2$
      and $|G_j(x_j)|(ave + 4\frac{\epsilon}{h} + 4\epsilon) \leq h\min(|G_j(x_{j-1})|, |G_j(x_{j+1})|)$
      and $((4\frac{\epsilon}{h} \leq min(G_j'(x_{j-1}), G_j'(x_j), G_j'(x_{j+1}))$ or $-4\frac{\epsilon}{h} \geq max(G_j'(x_{j-1}), G_j'(x_j), G_j'(x_{j+1}))))$
      and $|G_j(x_{j+1})| + 4\epsilon + 4\frac{\epsilon}{h} < |G_j'(x_{j+1})|$
      and $|G_j(x_{j-1})| + 4\epsilon + 4\frac{\epsilon}{h} < |G_j'(x_{j-1})|)$  then
            there is a jump at $x_j$
            $nn = 1$
      endif
endif
if  ($nn \neq 1$ and $A$ singularity can exist at $(x_{j-1}, x_j)$ adapted stencil selection)
  then
      if  $(G_j'(x_j)G_j'(x_{j-1}) < -16(\frac{\epsilon}{h})^2$ and $\min(|G_j'(x_{j-1})|, |G_j'(x_j)|) \geq h^2 + 8\frac{\epsilon}{h}$
      and $((8\frac{\epsilon}{h} \leq \min(G_j''(x_{j-1}), G_j''(x_j), G_j''(x_{j+1}))$ or $-8\frac{\epsilon}{h} \geq \max(G_j''(x_{j-1}), G_j''(x_j), G_j''(x_{j+1}))))$
      and $|G_j'(x_j)| + 8\frac{\epsilon}{h} + 8\frac{\epsilon}{h^2} < |G_j''(x_j)|$
      and $|G_j'(x_{j-1})| + 8\frac{\epsilon}{h} + 8\frac{\epsilon}{h^2} < |G_j''(x_{j-1})|)$  then
            there is a corner in $(x_{j-1}, x_j)$
      elseif  $(G_j(x_j)G_j(x_{j-1}) < -4\epsilon^2$ and $\min(|G_j(x_{j-1})|, |G_j(x_{j+1})|) \geq h^2ave + 4\epsilon + 4\epsilon h$
      and $((4\epsilon \leq \min(G_j'(x_{j-1}), G_j'(x_j), G_j'(x_{j+1}))$ or $-4\epsilon \geq \max(G_j'(x_{j-1}), G_j'(x_j), G_j'(x_{j+1}))))$
      and $|G_j(x_j)| + 4\epsilon + 4\epsilon h < |G_j'(x_j)|$
      and $|G_j(x_{j-1})| + 4\epsilon + 4\epsilon h < |G_j'(x_{j-1})|)$  then
            there is a jump in $(x_{j-1}, x_j)$
      endif
endif
```

Table 3: Algorithm to detect singularities. Cell-average framework.

4.3 Numerical experiments and Conclusions

In this section we will introduce some plots of modified signals and pictures by random noise. Our algorithm detects the real singularities with a great noise.

Our scheme detects only true discontinuities. When the noise used is too big (for which the true discontinuities and those taken place by the noise have the same characteristics) our algorithm doesn't detect anything. Nevertheless, we can see in our examples that we can consider very large noises, more than the classical detectors.

In our experiments we use the idea of decreasing sequence to detect the discontinuities. After we check the measure of discontinuity and we decide the true singularities if the size is big enough with respect the noise. As we said before, we will consider the cell-average framework.

We start in 1-D with a jump and a corner. In figure 2, we plot the signals and in figure 3, the perturbation. We use 64 points, and the noise is lees than 0.4 and 0.01 respectively. In table 4 we can see the good resolution of the detector.

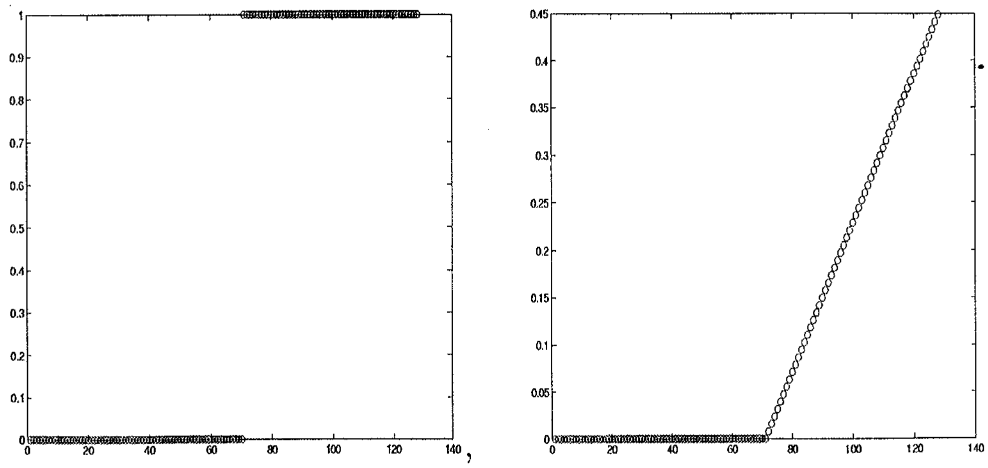

Fig. 2: left jump, right corner

Next, we apply our detector to images in 2-D. We start with a geometric figure captured with noise and finally we analyze a real image.

In figure 4, we consider a geometric picture without noise, and in figure 5 a

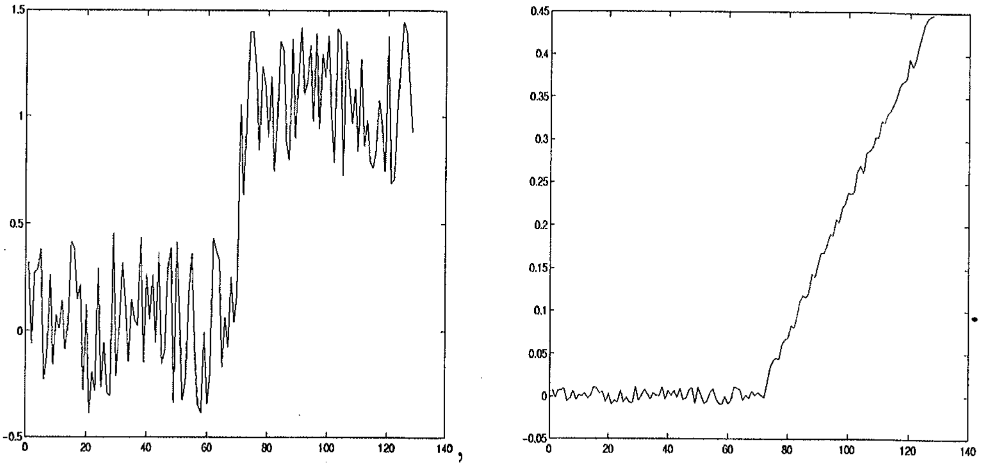

Fig. 3: left jump with noise, right corner with noise

	fig.2 left	fig.2 right	fig.3 left	fig.3 right
cell	$(35, 36)$	$(35, 36)$	$(35, 36)$	$(35, 36)$
location	0.547	0.547	0.499	0.510
size	1.	1.	1.	1.1
type	jump	corner	jump	corner

Table 4: $n = 64$, cell-average

perturbation of 4 with a noise lees than 10 is considered. We detect all the jumps, in table 5 we display their sizes (the real sizes are 50 in all the cases).

first jump	second jump	third jump
65	60	50
fourth jump	fifth jump	sixth jump
55	65	65

Table 5: $n = 256$, cell-average, noise=10

Finally, we work with a real image. In figures 6-9 we plot our experiments.

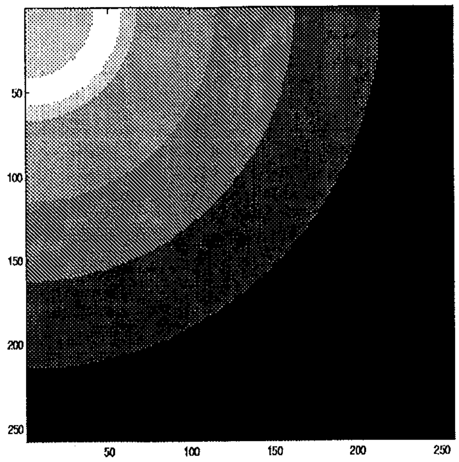

Fig. 4: noise=0

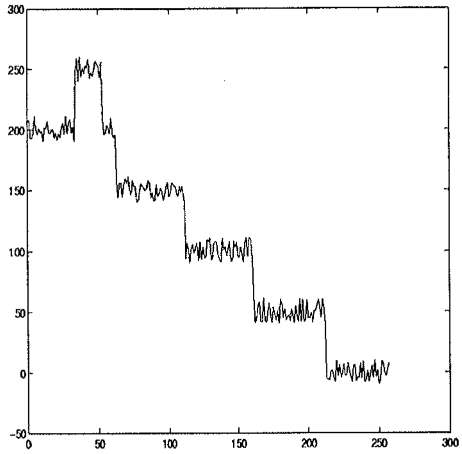

Fig. 5: section with noise

In figure 7 we can see the complicate structure of the real images. Our detector seems to work very well.

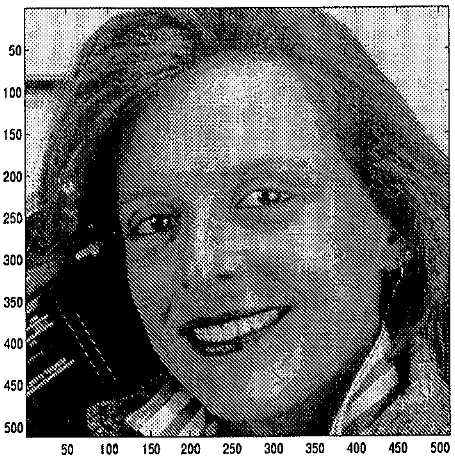

Fig. 6: real image

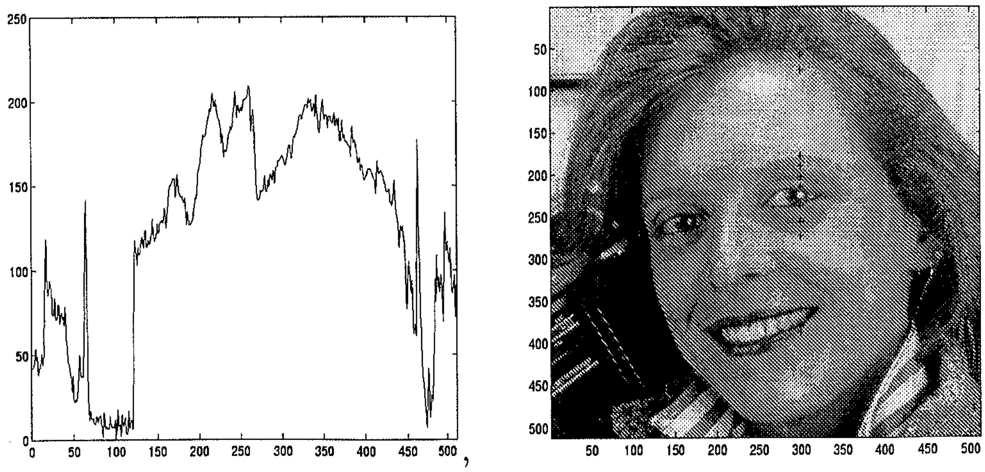

Fig. 7: left section-column=300, right detection, noise=0, cell-average

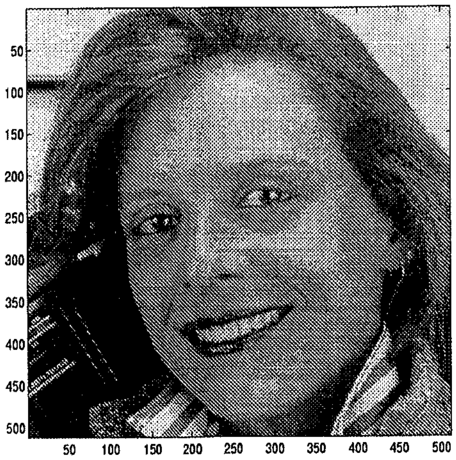

Fig. 8: noise=10

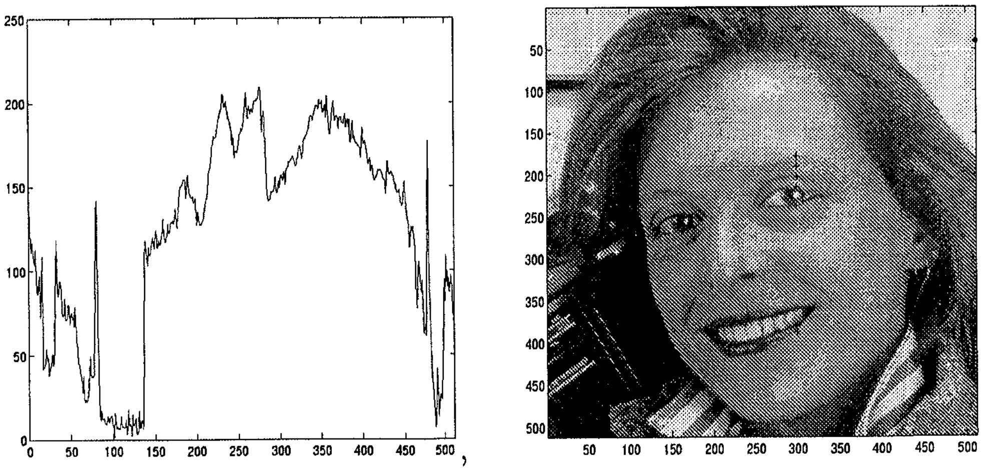

Fig. 9: left section-column=300, right detection,noise=10, cell-average

5 Multiresolution Analysis with Error Control in 2-D

A discrete sequence f^L is encoded to produce a multi-scale representation of its information contents, $(f^0, e^1, e^2, \ldots, e^L)$; this representation is then processed and the end result of this step is a modified multi-scale representation $(\hat{f}^0, \hat{e}^1, \hat{e}^2, \ldots, \hat{e}^L)$ which is *close* to the original one, i.e. such that (in some norm)

$$||\hat{f}^0 - f^0|| \leq \epsilon_0 \qquad ||\hat{e}^k - e^k|| \leq \epsilon_k \quad 1 \leq k \leq L,$$

where the truncation parameters $\epsilon_0, \epsilon_1, \ldots, \epsilon_L$ are chosen according to some criteria specified by the user. After decoding the processed representation, we obtain a discrete set $\hat{f}^L$ which is expected to be *close* to the original discrete set f^L. In order for this to be true, some form of stability is needed, i.e. we must require that

$$||\hat{f}^L - f^L|| \leq \sigma(\epsilon_0, \epsilon_1, \ldots, \epsilon_L)$$

where $\sigma(\cdot, \ldots, \cdot)$ satisfies

$$\lim_{\epsilon_l \to 0,\ 0 \leq l \leq L} \sigma(\epsilon_0, \epsilon_1, \ldots, \epsilon_L) = 0.$$

The stability analysis for linear prediction processes can be carried out using tools coming from wavelet theory, subdivision schemes and functional analysis (see [33], [9]), however none of these techniques is applicable in general when the prediction process is nonlinear.

In the nonlinear case, stability can be ensured by modifying the encoding algorithm. The idea of a modified-encoding to deal with nonlinear multiresolution schemes is due to Harten; one dimensional algorithms in several settings can be found in [31], [10], [8]. The goal of a modified-encoding procedure is to keep track of the accumulation error in processing the values in the multi-scale representation.

The aim of this section is to present two-dimensional multiresolution algorithms that ensure stability in the case of nonlinear prediction processes. In [4] we develop the stability in 2-D for the case of tensor product. Nevertheless, this is only one of the possible cases (the most natural way to extend the 1-D case). Different possibilities exist, for instance in [55] it is presented a method for image interpolation which adapts to the local characteristics of the image in order to facilitate perfectly smooth edges is outlined. A this feature in a visually optimized manner is nonlinear image enhancement is employed that extracts perceptually important details from the original image and uses these in order to improve the visual impression. Many different approaches have been used to derive these prediction operators, see [12], [16], [17], [41]. In section ten we present another approach.

In this work we will introduce a modified encoding for any reconstruction type.

5.1 Interpolatory MR analysis in 2-D

Let $X^k = \{x_i^k, y_j^k\}_{i,j=0}^{J_k}$, $J_k = 2^k J_0$, J_0 some integer, $x_{2i}^k = x_i^{k-1}$, $y_{2j}^k = y_j^{k-1}$ and $x_{2i-1}^k = (x_i^{k-1} + x_{i-1}^{k-1})/2$, $y_{2j-1}^k = (y_j^{k-1} + y_{j-1}^{k-1})/2$.

We consider

$$\mathcal{D}_k : \mathcal{C}([0,1]\times[0,1]) \longrightarrow V^k \qquad \bar{f}_{i,j}^k = (\mathcal{D}_k f)_{i,j} = f(x_i^k, y_j^k,), \quad 0 \le i,j \le J_k. \quad (26)$$

In this case, $\dim V^k = (J_k+1)\times(J_k+1)$ and the decimation operators are

$$\bar{f}_{i,j}^{k-1} = (D_k^{k-1}\bar{f}^k)_{i,j} = \bar{f}_{2i,2j}^k, \quad i,j = 0,1,\ldots,J_{k-1}.$$

Since $\bar{f}_{i,j}^{k-1} = \bar{f}_{2i,2j}^k$, we obtain $\mathcal{N}(D_k^{k-1}) = \{v^k \in V^k | v_{2i,2j}^k = 0\}$. Thus, if we denote by e^k the prediction errors, we will need to keep $e_{2i-1,2j-1}^k, e_{2i-1,2j}^k, e_{2i,2j-1}^k$ only.

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

$$\mathcal{R}_k : V^k \longrightarrow \mathcal{C}([0,1]\times[0,1]); \qquad \mathcal{D}_k\mathcal{R}_k\bar{f}^k = \bar{f}^k, \tag{27}$$

which means

$$(\mathcal{R}_k\bar{f}^k)(x_i^k, y_j^k) = \bar{f}_{i,j}^k = f(x_i^k, y_j^k). \tag{28}$$

Therefore, $\mathcal{R}_k$ should be a continuous function that interpolates the data $\bar{f}^k$ on at the grid points of X^k. Finally

$$P_{k-1}^k := \mathcal{D}_k\mathcal{R}_{k-1}. \tag{29}$$

The encoding and decoding algorithms take the following form:

Algorithm 5.1 $\mu(\bar{f}^L) = M\bar{f}^L$ *(Encoding)*

```
for $k = L, \dots, 1$
    for $i, j = 0, \dots, J_{k-1}$
        $\bar{f}_{i,j}^{k-1} = \bar{f}_{2i,2j}^k$
    end
    for $i = 1, \dots, J_{k-1}$
    for $j = 0, \dots, J_{k-1}$
        $f_{2i-1,2j}^P = (P_{k-1}^k\bar{f}^{k-1})_{2i-1,2j}$
        $e_{2i-1,2j}^k = \bar{f}_{2i-1,2j}^k - f_{2i-1,2j}^P$
    end
    end
    for $i = 0, \dots, J_{k-1}$
    for $j = 1, \dots, J_{k-1}$
        $f_{2i,2j-1}^P = (P_{k-1}^k\bar{f}^{k-1})_{2i,2j-1}$
        $e_{2i,2j-1}^k = \bar{f}_{2i,2j-1}^k - f_{2i,2j-1}^P$
    end
    end
    for $i, j = 1, \dots, J_{k-1}$
        $f_{2i-1,2j-1}^P = (P_{k-1}^k\bar{f}^{k-1})_{2i-1,2j-1}$
        $e_{2i-1,2j-1}^k = \bar{f}_{2i-1,2j-1}^k - f_{2i-1,2j-1}^P$
    end
end
```

$M^M \bar{f}^L = \{\bar{f}^0, e^1, \ldots, e^L\}$

Algorithm 5.2 $\bar{f}^L = M^{-1}\mu(\bar{f}^L)$ *(Decoding)*

```
for $k = 1, \ldots, L$
    for $i, j = J_{k-1}, \ldots, 0$
        $\bar{f}^k_{2i,2j} = \bar{f}^{k-1}_{i,j}$
    end
    for $i = J_{k-1}, \ldots, 1$
    for $j = J_{k-1}, \ldots, 0$
        $f^P_{2i-1,2j} = (P^k_{k-1}\bar{f}^{k-1})_{2i-1,2j}$
        $\bar{f}^k_{2i-1,2j} = e^k_{2i-1,2j} + f^P_{2i-1,2j}$
    end
    end
    for $i = J_{k-1}, \ldots, 0$
    for $j = J_{k-1}, \ldots, 1$
        $f^P_{2i,2j-1} = (P^k_{k-1}\bar{f}^{k-1})_{2i,2j-1}$
        $\bar{f}^k_{2i,2j-1} = e^k_{2i,2j-1} + f^P_{2i,2j-1}$
    end
    end
    for $i, j = J_{k-1}, \ldots, 1$
        $f^P_{2i-1,2j-1} = (P^k_{k-1}\bar{f}^{k-1})_{2i-1,2j-1}$
        $\bar{f}^k_{2i-1,2j-1} = e^k_{2i-1,2j-1} + f^P_{2i-1,2j-1}$
    end
end
```

5.2 Cell Average MR analysis in 2-D

In this case we have

$$\mathcal{D}_k : L^1([0,1] \times [0,1]) \longrightarrow V^k, \tag{30}$$

$$\bar{f}^k_{i,j} = (\mathcal{D}_k f)_{i,j} = \frac{1}{h_k^2} \int_{x^k_{i-1}}^{x^k_i} \int_{y^k_{j-1}}^{y^k_j} f(x,y)dydx, \quad 1 \leq i, j \leq J_k; \tag{31}$$

where $L^1([0,1]\times[0,1])$ is the space of absolutely integrable functions in $[0,1]\times[0,1]$ and the set of nested grids X^k is as in section 5.1. This analysis turns out to be appropriate for data compression of discontinuous, piecewise smooth signals.

It is sufficient to consider weighted averages $\bar{f}^k_{i,j}$ for $1 \leq i,j \leq J_k$ since these contain information on f over $[0,1]\times[0,1]$. Thus, V^k is the space of sequences with $J_k \times J_k$ components.

Moreover

$$\bar{f}^{k-1}_{i,j} = (D^{k-1}_k \bar{f}^k)_{i,j} = \frac{1}{4}(\bar{f}^k_{2i-1,2j-1} + \bar{f}^k_{2i-1,2j} + \bar{f}^k_{2i,2j-1} + \bar{f}^k_{2i,2j})$$

$i,j = 1,2,\ldots,J_{k-1}$.

On the other hand, since

$$0 = (D^{k-1}_k e^k)_{i,j} = (e^k_{2i-1,2j-1} + e^k_{2i-1,2j} + e^k_{2i,2j-1} + e^k_{2i,2j})/4$$

we will keep $e^k_{2i-1,2j-1}, e^k_{2i-1,2j}, e^k_{2i,2j-1}$ only.

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

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

$$(\mathcal{D}_k\mathcal{R}_k\bar{f}^k)_{i,j} = \frac{1}{h_k^2}\int_{x^k_{i-1}}^{x^k_i}\int_{y^k_{j-1}}^{y^k_j}(\mathcal{R}_k\bar{f}^k)(x,y)dxdy = \bar{f}^k_{i,j}.$$

That is, $\mathcal{R}_k\bar{f}^k(x,y)$ has to be a function in $L^1([0,1]\times[0,1])$ whose mean value on the (i,j)-th cell coincides with $\bar{f}^k_{i,j}$, $\forall i,j$, and $P^k_{k-1} := \mathcal{D}_k\mathcal{R}_{k-1}$.

The multiresolution transform and its inverse are now

Algorithm 5.3 $\mu(\bar{f}^L) = M\bar{f}^L$ *(Encoding)*

```
for $k = L,\ldots,1$
    for $i,j = 1,\ldots,J_{k-1}$
```

```
        $\bar{f}^{k-1}_{i,j} = \frac{1}{4}(\bar{f}^k_{2i-1,2j-1} + \bar{f}^k_{2i-1,2j} + \bar{f}^k_{2i,2j-1} + \bar{f}^k_{2i,2j})$
    end
    for $i, j = 1, \ldots, J_{k-1}$
        $f^P_{2i-1,2j-1} = (P^k_{k-1}\bar{f}^{k-1})_{2i-1,2j-1}$
        $e^k_{2i-1,2j-1} = \bar{f}^k_{2i-1,2j-1} - f^P_{2i-1,2j-1}$
        $f^P_{2i-1,2j} = (P^k_{k-1}\bar{f}^{k-1})_{2i-1,2j}$
        $e^k_{2i-1,2j} = \bar{f}^k_{2i-1,2j} - f^P_{2i-1,2j}$
        $f^P_{2i,2j-1} = (P^k_{k-1}\bar{f}^{k-1})_{2i,2j-1}$
        $e^k_{2i,2j-1} = \bar{f}^k_{2i,2j-1} - f^P_{2i,2j-1}$
    end
end
```

$M^M \bar{f}^L = \{\bar{f}^0, e^1, \ldots, e^L\}$

Algorithm 5.4 $\bar{f}^L = M^{-1}\mu(\bar{f}^L)$ *(Decoding)*

```
for $k = 1, \ldots, L$
    for $i, j = J_{k-1}, \ldots, 1$
        $f^P_{2i-1,2j-1} = (P^k_{k-1}\bar{f}^{k-1})_{2i-1,2j-1}$
        $\bar{f}^k_{2i-1,2j-1} = f^P_{2i-1,2j-1} + e^k_{2i-1,2j-1}$
        $f^P_{2i-1,2j} = (P^k_{k-1}\bar{f}^{k-1})_{2i-1,2j}$
        $\bar{f}^k_{2i-1,2j} = f^P_{2i-1,2j} + e^k_{2i-1,2j}$
        $f^P_{2i,2j-1} = (P^k_{k-1}\bar{f}^{k-1})_{2i,2j-1}$
        $\bar{f}^k_{2i,2j-1} = f^P_{2i,2j-1} + e^k_{2i,2j-1}$
        $\bar{f}^k_{2i,2j} = 4\bar{f}^{k-1}_{i,j} - \bar{f}^k_{2i-1,2j-1} - \bar{f}^k_{2i-1,2j} - \bar{f}^k_{2i,2j-1}$
    end
end
```

5.3 Multiresolution-based compression schemes with error-control

Multiresolution representations lead naturally to data-compression algorithms. The simplest data compression procedure is obtained by setting to zero all scale coefficients which fall below a prescribed tolerance. Let us denote

$$(\hat{e}^k)_{i,j} = \ \mathbf{tr}(e^k_{i,j};\epsilon_k) = \begin{cases} 0 & |e^k_{i,j}| \le \epsilon_k \\ e^k_{i,j} & \text{otherwise} \end{cases} \tag{32}$$

and refer to this operation as truncation. This type of data compression is used primarily to reduce the "dimensionality" of the data. A different strategy, which is used to reduce the digital representation of the data is "quantization", which can be modeled by

$$(\hat{e}^k)_{i,j} = \mathbf{qu}(e^k_{i,j};\epsilon_k) = 2\epsilon_k \cdot \ \text{round}\ \left[\frac{e^k_{i,j}}{2\epsilon_k}\right], \tag{33}$$

where round $[\cdot]$ denotes the integer obtained by rounding. For example, if $|e^k_{i,j}| \le 256$ and $\epsilon_k = 4$ then we can represent $e^k_{i,j}$ by an integer which is not larger than 32 and commit a maximal error of 4. Observe that if $|e^k_{i,j}| < \epsilon_k \Rightarrow \mathbf{qu}(e^k_{i,j};\epsilon_k) = 0$ and that in both cases

$$|e^k_{i,j} - \hat{e}^k_{i,j}| \le \epsilon_k. \tag{34}$$

By applying the inverse multiresolution transform to the compressed representation, we obtain $\hat{f}^L = M^{-1}\{\bar{f}^0, \hat{e}^1, \ldots, \hat{e}^L\}$, an approximation to the original signal $\bar{f}^L$. We expect the information contents of $\hat{f}^L$ to be very close to those of the original signal $\bar{f}^L$, and in order for this to be true, the stability of the multiresolution scheme with respect to perturbations is essential. Studying the effect of using $\hat{e}^k_{i,j}$ instead of $e^k_{i,j}$ in the input of M^{-1} is equivalent to studying the effect of a perturbation in the scale coefficients in the outcome of the inverse multiresolution transform.

Given a discrete sequence $\bar{f}^L$ and a tolerance level ϵ for accuracy, our task is to come up with a compressed representation

$$\{\bar{f}^0, \hat{e}^1, \ldots, \hat{e}^L\} \tag{35}$$

such that if $\hat{f}^L = M^{-1}\{\bar{f}^0, \hat{e}^1, \ldots, \hat{e}^L\}$, we have

$$\| \bar{f}^L - \hat{f}^L \| \leq C\epsilon \tag{36}$$

for an appropriate norm.

As observed by Harten [31], one possible way to accomplish this goal is to modify the encoding procedure in such a way that the modification allows us to keep track of the cumulative error and truncate accordingly.

In what follows we present a two-dimensional extension of the one dimensional algorithms in [31], [8] and the two dimensional tensor product in [4]. Given a tolerance level ϵ, the outcome of the modified encoding procedure is a compressed representation (35) satisfying (36). This enables us to specify the desired level of accuracy in the decompressed signal. A modified encoding procedure is designed keeping in mind the particular decoding procedure to be used.

We need the following definitions:

$$\left\|\bar{f}^k\right\|_\infty = \sup_{i,j}|\bar{f}^k_{i,j}|; \quad \left\|\bar{f}^k\right\|_1 = \frac{1}{J_k^2}(\sum_{i,j}|\bar{f}^k_{i,j}|); \quad \left\|\bar{f}^k\right\|_2^2 = \frac{1}{J_k^2}(\sum_{i,j}|\bar{f}^k_{i,j}|^2)$$

where $\bar{f}^k = \{\bar{f}^k_{ij}\}$.

We will consider truncation, but the algorithms are identical for another compression process.

5.4 Error-control algorithms

Algorithm 5.5 *Encoding for point values*

```
for $k = L, \ldots, 1$
    for $i, j = 0, \ldots, J_{k-1}$
        $\bar{f}^{k-1}_{i,j} = \bar{f}^k_{2i,2j}$
    end
```

```
end
Set $\hat{f}^0 = \bar{f}^0$
for $k = 1, \ldots, L$
    for $i = 1, \ldots, J_{k-1}$
    for $j = 0, \ldots, J_{k-1}$
        $\hat{e}^k_{2i-1,2j} = \mathrm{tr}(\bar{f}^k_{2i-1,2j} - (P^k_{k-1}\hat{f}^{k-1})_{2i-1,2j}, \epsilon_k)$
        $\hat{f}^k_{2i-1,2j} = (P^k_{k-1}\hat{f}^{k-1})_{2i-1,2j}, +\hat{e}^k_{2i-1,2j}$
    end
    end
    for $i = 0, \ldots, J_{k-1}$
    for $j = 1, \ldots, J_{k-1}$
        $\hat{e}^k_{2i,2j-1} = \mathrm{tr}(\bar{f}^k_{2i,2j-1} - (P^k_{k-1}\hat{f}^{k-1})_{2i,2j-1}, \epsilon_k)$
        $\hat{f}^k_{2i,2j-1} = (P^k_{k-1}\hat{f}^{k-1})_{2i,2j-1} + \hat{e}^k_{2i,2j-1}$
end
end
    for $i = 1, \ldots, J_{k-1}$
    for $j = 1, \ldots, J_{k-1}$
        $\hat{e}^k_{2i-1,2j-1} = \mathrm{tr}(\bar{f}^k_{2i-1,2j-1} - (P^k_{k-1}\hat{f}^{k-1})_{2i-1,2j-1}, \epsilon_k)$
        $\hat{f}^k_{2i-1,2j-1} = (P^k_{k-1}\hat{f}^{k-1})_{2i-1,2j-1} + \hat{e}^k_{2i-1,2j-1}$
    end
    end
    for $i = 0, \ldots, J_{k-1}$
    for $j = 0, \ldots, J_{k-1}$
        $\hat{f}^k_{2i,2j} = \hat{f}^{k-1}_{i,j}$
    end
    end
end
```

We denote $\tilde{e}^k_{i,j} := |\bar{f}^k_{i,j} - (P^k_{k-1}\hat{f}^{k-1})_{i,j}|$

Algorithm 5.6 *Encoding for cell-average*

```
for $k = L, \ldots, 1$
    for $i, j = 1, \ldots, J_{k-1}$
        $\bar{f}^{k-1}_{i,j} = \frac{1}{4}(\bar{f}^k_{2i-1,2j-1} + \bar{f}^k_{2i-1,2j} + \bar{f}^k_{2i,2j-1} + \bar{f}^k_{2i,2j})$
    end
end
Set $\hat{f}^0 = \bar{f}^0$
```

```
for $k = 1, \ldots, L$
    for $i = 1, \ldots, J_{k-1}$
    for $j = 1, \ldots, J_{k-1}$
        $f^P_{2i-1,2j-1} = (P^k_{k-1}\hat{f}^{k-1})_{2i-1,2j-1}$
        $\hat{e}^k_{2i-1,2j-1} = \mathrm{tr}([\bar{f}^k_{2i-1,2j-1} - f^P_{2i-1,2j-1}] - [\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}], \epsilon_k)$
        $\hat{f}^k_{2i-1,2j-1} = f^P_{2i-1,2j-1} + \hat{e}^k_{2i-1,2j-1}$
        $f^P_{2i-1,2j} = (P^k_{k-1}\hat{f}^{k-1})_{2i-1,2j}$
        $\hat{e}^k_{2i-1,2j} = \mathrm{tr}([\bar{f}^k_{2i-1,2j} - f^P_{2i-1,2j}] - [\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}], \epsilon_k)$
        $\hat{f}^k_{2i-1,2j} = f^P_{2i-1,2j} + \hat{e}^k_{2i-1,2j}$
        $f^P_{2i,2j-1} = (P^k_{k-1}\hat{f}^{k-1})_{2i,2j-1}$
        $\hat{e}^k_{2i,2j-1} = \mathrm{tr}([\bar{f}^k_{2i,2j-1} - f^P_{2i,2j-1}] - [\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}], \epsilon_k)$
        $\hat{f}^k_{2i,2j-1} = f^P_{2i,2j-1} + \hat{e}^k_{2i,2j-1}$
        $\hat{f}^k_{2i,2j} = 4\hat{f}^{k-1}_{i,j} - \hat{f}^k_{2i-1,2j-1} - \hat{f}^k_{2i-1,2j} - \hat{f}^k_{2i,2j-1}$
    end
    end
end
```

We denote
$\tilde{e}^k_{2i,2j-1} = ||[\bar{f}^k_{2i,2j-1} - f^P_{2i,2j-1}] - [\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}]||$
$\tilde{e}^k_{2i-1,2j-1} = ||[\bar{f}^k_{2i-1,2j-1} - f^P_{2i-1,2j-1}] - [\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}]||$
$\tilde{e}^k_{2i-1,2j} = ||[\bar{f}^k_{2i-1,2j} - f^P_{2i-1,2j}] - [\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}]||$

REMARK 5.1 *We will denote* $e^k_{i,j}(1) := e^k_{2i-1,2j-1}$, $e^k_{i,j}(2) := e^k_{2i-1,2j}$, $e^k_{i,j}(3) := e^k_{2i,2j-1}$.

Proposition 5.1 *Given a discrete sequence $\bar{f}^L$, with the modified encoding algorithm for the point value framework in 2-d (Algorithm 5.5) we obtain a multiresolution representation $M\bar{f}^L = \{\bar{f}^0, \hat{e}^1, \ldots, \hat{e}^L\}$ such that if we apply the decoding algorithm we obtain $\hat{f}^L$ satisfying:*

$$||\bar{f}^L - \hat{f}^L||_\infty \quad = \quad \max_k(||\bar{f}^0 - \hat{f}^0||_\infty, |||\tilde{e}^k - \hat{e}^k|||_\infty) \tag{37}$$

$$\begin{aligned}
||\bar{f}^L - \hat{f}^L||_1 &= \frac{1}{4^L}||\bar{f}^0 - \hat{f}^0||_1 + \sum_{k=1}^{L} \frac{1}{4^{L-k+1}}|||\tilde{e}^k - \hat{e}^k|||_1 && (38)\\
||\bar{f}^L - \hat{f}^L||_2^2 &= \frac{1}{4^L}||\bar{f}^0 - \hat{f}^0||_2^2 + \sum_{k=1}^{L} \frac{1}{4^{L-k+1}}|||\tilde{e}^k - \hat{e}^k|||_2^2 && (39)
\end{aligned}$$

where

$$\begin{aligned}
|||\tilde{e}^k - \hat{e}^k|||_\infty &= \max(||\tilde{e}^k(1) - \hat{e}^k(1)||_\infty, ||\tilde{e}^k(2) - \hat{e}^k(2)||_\infty,\\
&\quad ||\tilde{e}^k(3) - \hat{e}^k(3)||_\infty),\\
|||\tilde{e}^k - \hat{e}^k|||_1 &= ||\tilde{e}^k(1) - \hat{e}^k(1)||_1 + ||\tilde{e}^k(2) - \hat{e}^k(2)||_1 +\\
&\quad ||\tilde{e}^k(3) - \hat{e}^k(3)||_1,\\
|||\tilde{e}^k - \hat{e}^k|||_2^2 &= ||\tilde{e}^k(1) - \hat{e}^k(1)||_2^2 + ||\tilde{e}^k(2) - \hat{e}^k(2)||_2^2 +\\
&\quad ||\tilde{e}^k(3) - \hat{e}^k(3)||_2^2.
\end{aligned}$$

Proof

From the encoding algorithm we obtain:

$$\begin{aligned}
\bar{f}^k_{2i-1,2j} - \hat{f}^k_{2i-1,2j} &= \tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}\\
\bar{f}^k_{2i,2j-1} - \hat{f}^k_{2i,2j-1} &= \tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1}\\
\bar{f}^k_{2i-1,2j-1} - \hat{f}^k_{2i-1,2j-1} &= \tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}\\
\bar{f}^k_{2i,2j} - \hat{f}^k_{2i,2j} &= \bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}.
\end{aligned}$$

Then

$$\begin{aligned}
||\bar{f}^k - \hat{f}^k||_\infty &= \sup_{i,j} |\bar{f}^k_{i,j} - \hat{f}^k_{i,j}|\\
&= \sup_{i,j}(|\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}|, |\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}|,
\end{aligned}$$

$$
\begin{aligned}
& |\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}|, |\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1}|) \\
= \; & \max(||\bar{f}^{k-1} - \hat{f}^{k-1}||_\infty, |||\tilde{e}^k - \hat{e}^k|||_\infty)
\end{aligned}
$$

and we obtain (37).

Since $J_k = 2J_{k-1}$, taking $p = 1$ (or 2),

$$
\begin{aligned}
||\bar{f}^k - \hat{f}^k||_p^p &= \frac{1}{J_k^2} \sum_{i,j}^{J_k} |\bar{f}^k_{i,j} - \hat{f}^k_{i,j}|^p \\
&= \frac{1}{4J_{k-1}^2} \sum_{i,j}^{J_{k-1}} (|\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}|^p + |\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}|^p + \\
&\quad |\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}|^p + |\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1}|^p) \\
&= \frac{1}{4} ||\bar{f}^{k-1} - \hat{f}^{k-1}||_p^p + \frac{1}{4} |||\tilde{e}^k - \hat{e}^k|||_p^p
\end{aligned}
$$

which proves (38) and (39).

□

It is absolutely trivial then to prove the following corollary.

Corollary 5.1 *Consider the error control multiresolution scheme described in proposition 5.1, and a processing strategy for the scale coefficients such that*

$$
||\tilde{e}^k(l) - \hat{e}^k(l)||_p \le \epsilon_k \quad l = 1, 2, 3, \quad p = \infty, 1, \; or \; 2 \tag{40}
$$

Then we have

$$
\begin{aligned}
||\bar{f}^L - \hat{f}^L||_\infty &\le \max_k(\epsilon_k, ||\bar{f}^0 - \hat{f}^0||_\infty) \\
||\bar{f}^L - \hat{f}^L||_1 &\le \frac{1}{4^L} ||\bar{f}^0 - \hat{f}^0||_1 + 3 \sum_{k=1}^{L} \frac{\epsilon_k}{4^{L-k+1}} \\
||\bar{f}^L - \hat{f}^L||_2^2 &\le \frac{1}{4^L} ||\bar{f}^0 - \hat{f}^0||_2^2 + 3 \sum_{k=1}^{L} \frac{\epsilon_k^2}{4^{L-k+1}}
\end{aligned}
$$

In particular, if we assume that $||\bar{f}^0 - \hat{f}^0|| = 0$ *and we consider*

$$\epsilon_k = \frac{\epsilon}{q^{L-k+1}} \qquad q \geq 1 \tag{41}$$

we obtain

$$\begin{aligned} ||\bar{f}^L - \hat{f}^L||_\infty &\leq \frac{\epsilon}{q}, \\ ||\bar{f}^L - \hat{f}^L||_1 &\leq \frac{3\epsilon}{4q-1}, \\ ||\bar{f}^L - \hat{f}^L||_2^2 &\leq \frac{3\epsilon^2}{4q^2-1}. \end{aligned}$$

These bounds become

$$||\bar{f}^L - \hat{f}^L||_p \leq \epsilon, \quad p = 1, 2, \infty \tag{42}$$

for $q = 1$ *and*

$$||\bar{f}^L - \hat{f}^L||_\infty \leq \frac{\epsilon}{2}; \quad ||\bar{f}^L - \hat{f}^L||_1 \leq \frac{3}{7}\epsilon; \quad or \quad ||\bar{f}^L - \hat{f}^L||_2 \leq \frac{\epsilon}{\sqrt{5}} \tag{43}$$

for $q = 2$.

Proposition 5.2 *Given a discrete sequence* $\bar{f}^L$*, with the modified encoding algorithm for the cell-average framework in 2-d (Algorithm 5.6) we obtain a multiresolution representation* $M\bar{f}^L = \{\bar{f}^0, \hat{e}^1, \ldots, \hat{e}^L\}$ *such that if we apply the decoding algorithm we obtain* $\hat{f}^L$ *satisfying:*

$$||\bar{f}^L - \hat{f}^L||_\infty \leq ||\bar{f}^0 - \hat{f}^0||_\infty + 3\sum_{k=1}^{L} |||\tilde{e}^k - \hat{e}^k|||_\infty \tag{44}$$

$$||\bar{f}^L - \hat{f}^L||_1 \leq ||\bar{f}^0 - \hat{f}^0||_1 + \frac{1}{2}\sum_{k=1}^{L} |||\tilde{e}^k - \hat{e}^k|||_1 \tag{45}$$

$$||\bar{f}^L - \hat{f}^L||_2^2 = ||\bar{f}^0 - \hat{f}^0||_2^2 + \frac{1}{4}\sum_{k=1}^{L} |||\tilde{e}^k - \hat{e}^k|||_2^2 + \frac{1}{4}\sum_{k=1}^{L} ||\tilde{E}^k - \hat{E}^k||_2^2 \tag{46}$$

where

$$\begin{aligned} |||\tilde{e}^k - \hat{e}^k|||_\infty &= \max(||\tilde{e}^k(1) - \hat{e}^k(1)||_\infty, ||\tilde{e}^k(2) - \hat{e}^k(2)||_\infty, \\ &\quad ||\tilde{e}^k(3) - \hat{e}^k(3)||_\infty), \\ |||\tilde{e}^k - \hat{e}^k|||_1 &= ||\tilde{e}^k(1) - \hat{e}^k(1)||_1 + ||\tilde{e}^k(2) - \hat{e}^k(2)||_1 + \\ &\quad ||\tilde{e}^k(3) - \hat{e}^k(3)||_1, \\ |||\tilde{e}^k - \hat{e}^k|||_2^2 &= ||\tilde{e}^k(1) - \hat{e}^k(1)||_2^2 + ||\tilde{e}^k(2) - \hat{e}^k(2)||^2 + \\ &\quad ||\tilde{e}^k(3) - \hat{e}^k(3)||_2^2, \end{aligned}$$

and

$$\tilde{E}^k_{i,j} - \hat{E}^k_{i,j} = ((\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}) + (\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}) + (\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1})).$$

Proof

From the encoding algorithm we obtain:

$$\begin{aligned} \bar{f}^k_{2i-1,2j} - \hat{f}^k_{2i-1,2j} &= \bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j} + (\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}) \\ \bar{f}^k_{2i,2j-1} - \hat{f}^k_{2i,2j-1} &= \bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j} + (\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1}) \\ \bar{f}^k_{2i-1,2j-1} - \hat{f}^k_{2i-1,2j-1} &= \bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j} + (\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}) \\ \bar{f}^k_{2i,2j} - \hat{f}^k_{2i,2j} &= (4\bar{f}^{k-1}_{i,j} - \bar{f}^k_{2i-1,2j-1} - \bar{f}^k_{2i-1,2j} - \bar{f}^k_{2i,2j-1}) \\ &\quad -(4\hat{f}^{k-1}_{i,j} - \hat{f}^k_{2i-1,2j-1} - \hat{f}^k_{2i-1,2j} - \hat{f}^k_{2i,2j-1}) \\ &= \bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j} - (\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}) \\ &\quad -(\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1}) - (\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}). \end{aligned}$$

Then

$$||\bar{f}^k - \hat{f}^k||_\infty = \max(\bar{f}^k_{2i,2j} - \hat{f}^k_{2i,2j}, \bar{f}^k_{2i-1,2j} - \hat{f}^k_{2i-1,2j},$$

$$\begin{aligned} & \bar{f}^k_{2i-1,2j-1} - \hat{f}^k_{2i-1,2j-1}, \bar{f}^k_{2i,2j-1} - \hat{f}^k_{2i,2j-1}) \\ \leq \;& ||\bar{f}^{k-1} - \hat{f}^{k-1}||_\infty + 3\max(||\tilde{e}^{(}1) - \hat{e}^k(1)||_\infty, \\ & ||\tilde{e}^k(2) - \hat{e}^k(2)||_\infty, ||\tilde{e}^k(3) - \hat{e}^k(3)||_\infty) \end{aligned}$$

and we obtain (44).

Since $J_k = 2J_{k-1}$, taking $p = 1$,

$$|\bar{f}^k_{2i,2j} - \hat{f}^k_{2i,2j}| + |\bar{f}^k_{2i-1,2j} - \hat{f}^k_{2i-1,2j}| + |\bar{f}^k_{2i-1,2j-1} - \hat{f}^k_{2i-1,2j-1}| + |\bar{f}^k_{2i,2j-1} - \hat{f}^k_{2i,2j-1}|$$

$$\begin{aligned} \leq 4 \;\; & |\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}| + 2(|\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}| + |\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}| \\ & + |\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1}|) \end{aligned}$$

From the previous one and from the definition of $||\cdot||_1$-norm we obtain

$$\begin{aligned} ||\bar{f}^k - \hat{f}^k||_1 \;\; & \leq \;\; ||\bar{f}^{k-1} - \hat{f}^{k-1}||_1 + \frac{1}{2}(||\tilde{e}^k(1) - \hat{e}^k(1)||_1 \\ & + \;\; ||\tilde{e}^k(2) - \hat{e}^k(2)||_1 + ||\tilde{e}^k(3) - \hat{e}^k(3)||_1) \end{aligned}$$

which proves (45).

Finally, from

$$\begin{aligned} & (\bar{f}^k_{2i,2j} - \hat{f}^k_{2i,2j})^2 + (\bar{f}^k_{2i-1,2j} - \hat{f}^k_{2i-1,2j})^2 + (\bar{f}^k_{2i-1,2j-1} - \hat{f}^k_{2i-1,2j-1})^2 \\ + \;\; & (\bar{f}^k_{2i,2j-1} - \hat{f}^k_{2i,2j-1})^2 \\ = \;\; & \{(\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j})^2 - 2(\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j}) \\ \cdot \;\; & ((\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}) + (\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}) + (\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1})) \\ + \;\; & ((\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}) + (\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}) + (\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1}))^2\} \\ + \;\; & \{(\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j})^2 + 2(\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j})(\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}) \\ + \;\; & (\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j})^2\} \end{aligned}$$

$$\begin{aligned}
&+ \quad \{(\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j})^2 + 2(\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j})(\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}) \\
&+ \quad (\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1})^2\} \\
&+ \quad \{(\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j})^2 + 2(\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j})(\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1}) \\
&+ \quad (\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1})^2\} \\
&= \quad 4(\bar{f}^{k-1}_{i,j} - \hat{f}^{k-1}_{i,j})^2 + (\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j})^2 \\
&+ \quad (\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1})^2 + (\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1})^2 \\
&+ \quad ((\tilde{e}^k_{2i-1,2j} - \hat{e}^k_{2i-1,2j}) + (\tilde{e}^k_{2i-1,2j-1} - \hat{e}^k_{2i-1,2j-1}) + (\tilde{e}^k_{2i,2j-1} - \hat{e}^k_{2i,2j-1}))^2
\end{aligned}$$

we obtain

$$\begin{aligned}
||\bar{f}^k - \hat{f}^k||_2^2 &= ||\bar{f}^{k-1} - \hat{f}^{k-1}||_2^2 + \frac{1}{4}(||\tilde{e}^k(1) - \hat{e}^k(1)||_2^2 \\
&+ ||\tilde{e}^k(2) - \hat{e}^k(2)||_2^2 + ||\tilde{e}^k(3) - \hat{e}^k(3)||_2^2 \\
&+ ||\tilde{E}^k - \hat{E}^k||_2^2).
\end{aligned}$$

And the proposition has been proved.

□

It is absolutely trivial then to prove the following corollary.

Corollary 5.2 *Consider the error control multiresolution scheme described in proposition 5.2, and a processing strategy for the scale coefficients such that*

$$||\tilde{e}^k(l) - \hat{e}^k(l)||_p \leq \epsilon_k \quad l = 1,2,3, \quad p = \infty, 1, \ or\ 2 \tag{47}$$

Then we have

$$||\bar{f}^L - \hat{f}^L||_\infty \ \leq \ ||\bar{f}^0 - \hat{f}^0||_\infty + 3\sum_{k=1}^{L} \epsilon_k$$

$$\begin{aligned}
||\bar{f}^L - \hat{f}^L||_1 &\leq ||\bar{f}^0 - \hat{f}^0||_1 + \frac{3}{2}\sum_{k=1}^{L} \epsilon_k \\
||\bar{f}^L - \hat{f}^L||_2^2 &\leq ||\bar{f}^0 - \hat{f}^0||_2^2 + 3\sum_{k=1}^{L} \epsilon_k^2
\end{aligned}$$

In particular, if we assume that $||\bar{f}^0 - \hat{f}^0|| = 0$ *and we consider*

$$\epsilon_k = \frac{\epsilon}{q^{L-k+1}} \qquad q > 1 \tag{48}$$

we obtain

$$\begin{aligned}
||\bar{f}^L - \hat{f}^L||_\infty &\leq \frac{3\epsilon}{q-1}, \\
||\bar{f}^L - \hat{f}^L||_1 &\leq \frac{3}{2}\frac{\epsilon}{q-1}, \\
||\bar{f}^L - \hat{f}^L||_2^2 &\leq \frac{3\epsilon^2}{q^2-1}.
\end{aligned}$$

These bounds become

$$||\bar{f}^L - \hat{f}^L||_\infty \leq 3\epsilon; \quad ||\bar{f}^L - \hat{f}^L||_1 \leq \frac{3}{2}\epsilon; \quad or \quad ||\bar{f}^L - \hat{f}^L||_2 \leq \epsilon \tag{49}$$

for $q = 2$.

Proof

Notice you,

$$||\tilde{E}^k - \hat{E}^k||_2 \leq ||\tilde{e}^k(1) - \hat{e}^k(1)||_2 + ||\tilde{e}^k(2) - \hat{e}^k(2)||_2 + ||\tilde{e}^k(3) - \hat{e}^k(3)||_2$$

Then from (46) we obtain,

$$\begin{aligned}
||\bar{f}^L - \hat{f}^L||_2^2 &\leq ||\bar{f}^0 - \hat{f}^0||_2^2 + \frac{1}{4}\sum_{k=1}^{L} 3\epsilon_k^2 + \frac{1}{4}\sum_{k=1}^{L} 9\epsilon_k^2 \\
&= ||\bar{f}^0 - \hat{f}^0||_2^2 + 3\sum_{k=1}^{L} \epsilon_k^2
\end{aligned}$$

□

6 Nearly-Optimal Nonlinear Multiresolution Algorithms

A nonlinear multiresolution-packets is presented. The stability and the improvements of the new algorithm are studied. Our goal is to describe tools for adapting methods of analysis to various tasks occuring in harmonic analysis and signal processing. The main point of this presentation is that by choosing a representation, in which time, frequency and smoothness are suitably localized.

We would like to describe a method permitting efficient compression of a variety of signals such as sound and images. The method can use any linear or nonlinear multiresolution. In the first case, we can recover the biorthogonal wavelet-packets and the interpolating wavelet-packets, but in the second case a new algorithm is obtained.

The development of the theory of wavelets has closely followed its practical applications. The wavelet bases have poor frequency localization when the number of scales "L" is large. For some applications especially for signal processing, it is more convenient to have bases with better frequency localization. This will be provided by the wavelet-packets, which are obtained from wavelets associated with multiresolution analysis.

This idea can be made precise as well as generalized to all multiresolution with a tree structure. There are a denumerable number of ways to choose a decomposition of the signal. This flexibility of choosing the decomposition of the signal is well adapted for applications, where we can choose to decompose the details according to the available data. The multiresolution-packets framework allows to define the notion of a minimal decomposition that has proven to be an efficient procedure for data compression. The purpose here is to take benefit of this compression to represent accurately and economically a signal.

On the other hand, Harten's experience in numerical methods for hyperbolic systems of conservation laws, lead him to the conclusion that one has to "go nonlinear" when approximating functions with singularities. The idea of the section is to use this advantages in order to construct an efficient nonlinear multiresolution-packets. Our construction will be directed toward numerical applications. We will describe several discrete algorithms. We will show the correspondence between multiresolution-packets and coefficients computed from sampled signal. We will examine several compression methods, both linear and nonlinear.

There are several applications of these representations as: image analysis [57], compression data [19], adaptive methods for approximation of nonstationary partial differential equations [43],...

The concept of wavelet-packets has been introduced by R.R.Coifman et al. [19],[20] as a generalization of wavelet bases. It relies on the definition of a library of bases, the best bases is choosen so as to minimize some given entropy attached to the coefficients in each bases of the library.

Even in the linear case, it's obviously a nonlinear transformation (decomposition and reconstruction algorithms) to represent a signal in its own best multiresolution. In this case since the transformation is biorthogonal once the basis is choosen, compression via the best multiresolution is not drastically affected by the noise. Furthermore, for the nonlinear case we have to modify the direct algorithms in order to ensure stability. In this section we introduce some error-control algorithms for different multiresolutions.

The section is organized as follows: we will introduce the general multiresolution-packets. Special attention is paid to error-control in next subsection. And finally several numerical experiments and some conclusions are presented.

6.1 Multiresolution packets

In this section we shall introduce the general "Multiresolution packet". The same as the library of wavelet packet bases it is naturally organized as subsets of binary tree. This segmentation of signals into those dyadic intervals is better adapted to the frequency content.

The idea is to obtain the best decomposition of all the possible ones. We now define a cost function on sequence and search for its minimum over all representations in a library. For a given vector, their minima are the most efficient representation.

Definition 6.1 *A map $\mathcal{L}$ from sequences $\{x_j\}$ to R is called an additive information cost function if $\mathcal{L}(0) = 0$ and $\mathcal{L}(\{x_j\}) = \sum_j \mathcal{L}(x_j)$.*

Some useful examples of information cost include: a)Number above a threshold: set an arbitrary threshold ϵ and count the elements in the sequence x whose absolute value exceeds ϵ. b) Concentration in l^p norm $(p < 2)$, $\mathcal{L}(x) = ||x||_p$. c) Entropy, $\mathcal{L}(x) = -\sum_j p_j log p_j$ where $p_j = \frac{|x_j|^2}{||x||^2}$ and we set $p \; log p = 0$ if $p = 0$. d) Logarithm of energy, $\mathcal{L}(x) = \sum_j log|x_j|^2$. Here we will use the first possibility.

As the library is a tree, then we can find the best representation by induction on the number of scales. Denote by g_j^k the representation of vectors corresponding to the scale k, $j = 1, 2, \ldots, 2^{L-k}$, and by $\mathcal{B}_j^k$ the best representation for x. For $k = L$, $\mathcal{B}^L = g^L$. We construct $\mathcal{B}_j^{k-1}$ as follows:

$$\mathcal{B}_j^{k-1} = \begin{cases} g_j^{k-1} & \mathcal{L}(\mathcal{B}_j^k) > \mathcal{L}(g_{2j}^{k-1}) + \mathcal{L}(g_{2j+1}^{k-1}) \\ \mathcal{B}_{2j}^k + \mathcal{B}_{2j+1}^k & \text{otherwise} \end{cases} \tag{50}$$

In practice, we start with a vector of data $\bar{f}^k$, corresponding to any discretization of a certain function. We compute a step of the multiresolution algorithm, that is, $\bar{f}^{k-1}$ and the details $\bar{d}^k$. If the addition of the cost of these two new vectors is higher than it comes from $\bar{f}^k$ we do not consider the decomposition. On the other hand, if the cost is less then we carry out the decomposition. If it has been produced the latest case then we would repeat the process for these two new vectors ($\bar{f}^{k-1}$ and $\bar{d}^k$) independently. Anyhow, the decomposition is finished when one has arrived to the coarsest resolution level prescribed by the user.

Now, we will introduce the general algorithm for the case of point values. Let be $\bar{f}^L = (f(x_j^L))_j$ the $J_L + 1$ sequence corresponding to the point-values discretization at the finest resolution level L and P_{k-1}^k some prediction operators (interpolation process usually). Then we obtain the following encoding algorithm

Algorithm 6.1

```
for $k = L, \ldots, 1$
    for  $j = 0, \ldots, J_{k-1}$
        $\bar{f}_j^{k-1} = \bar{f}_{2j}^k$
    end
    for  $j = 1, \ldots, J_{k-1}$
        $f_{2j-1}^P = (P_{k-1}^k \bar{f}^{k-1})_{2j-1}$
        $\bar{d}_j^k = \bar{f}_{2j-1}^k - f_{2j-1}^P$
    end
        $e^k = MP(\bar{d}^k)$
end
  $M^M \bar{f}^L = \{\bar{f}^0, e^1, \ldots, e^L\}$
```

In the cell average framework, if $\bar{f}^L$ is a J_L sequence corresponding to the cell-average discretization at the finest resolution level L, $\bar{f}_j^k := \frac{1}{h_k}\int_{x_{j-1}^k}^{x_j^k} f(x)dx$ we will have

Algorithm 6.2

```
for  $k = L, \dots, 1$
    for  $j = 1, \dots, J_{k-1}$
        $\bar{f}_j^{k-1} = \frac{1}{2}(\bar{f}_{2j}^k + \bar{f}_{2j-1}^k)$
    end
    for  $j = 1, \dots, J_{k-1}$
        $f_{2j-1}^P = (P_{k-1}^k \bar{f}^{k-1})_{2j-1}$
        $\bar{d}_j^k = \bar{f}_{2j-1}^k - f_{2j-1}^P$
    end
        $e^k = MP(\bar{d}^k)$
end
    $M^M \bar{f}^L = \{\bar{f}^0, e^1, \dots, e^L\}$
```

REMARK 6.1 *MP indicates the multiresolution process, applied to all the details, with the selection of the best representation.*

6.2 Error-control algorithms

In this section we are going to study the stability concept in the multiresolution-packets framework. We have to consider modified algorithms. Next, we shall introduce some results of stability associated to the modified algorithms. We can consider different multiresolutions (linear and non linear) for the scales and the details. The modified algorithms will have a similar structure than the original algorithms introduced by Harten for the multiresolution framework [31].

Modified encoding procedure for the interpolatory framework:

Algorithm 6.3

```
for $k = L, \dots, 1$
    for $j = 0, \dots, J_{k-1}$
        $\bar{f}_j^{k-1} = \bar{f}_{2j}^k$
    end
end
Set $\hat{f}^0 = \bar{f}^0$
for $k = 1, \dots, L$
        $\hat{f}_0^k = \bar{f}_0^k$
    for $j = 1, \dots, J_{k-1}$
        $f_{2j-1}^P = (P_{k-1}^k \hat{f}^{k-1})_{2j-1}$
        $\bar{d}_j^k = \bar{f}_{2j-1}^k - f_{2j-1}^P$
    end
        $\hat{e}^k = tr(MP(\bar{d}^k, \epsilon_k))$
        $\hat{d}^k = (\hat{M}\hat{P})^{-1}\hat{e}^k$
    for $j = 1, \dots, J_{k-1}$
        $\hat{f}_{2j-1}^k = f_{2j-1}^P + \hat{d}_j^k$
        $\hat{f}_{2j}^k = \hat{f}_j^{k-1}$
    end
end
    $M^M \bar{f}^L = \{\bar{f}^0, \hat{e}^1, \dots, \hat{e}^L\}$
```

REMARK 6.2 *Whenever a parent node is of lower information cost than the children, we mark the parent. In the representation $\{\hat{e}^1, \dots, \hat{e}^L\}$ we have all the information, that is, the value of the details and the marks. With $(\hat{M}\hat{P})^{-1}$ we recover some approximation of the value $\bar{d}^k$.*

Modified encoding procedure for the cell average framework is:

Algorithm 6.4

```
for  $k = L, \dots, 1$
    for  $j = 1, \dots, J_{k-1}$
        $\bar{f}_j^{k-1} = \frac{1}{2}(\bar{f}_{2j}^k + \bar{f}_{2j-1}^k)$
    end
end
Set  $\hat{f}^0 = \bar{f}^0$
for  $k = 1, \dots, L$
    for  $j = 1, \dots, J_{k-1}$
        $f_{2j-1}^P = (P_{k-1}^k \hat{f}^{k-1})_{2j-1}$
        $\bar{d}_j^k = [\bar{f}_{2j-1}^k - f_{2j-1}^P] - [\bar{f}_j^{k-1} - \hat{f}_j^{k-1}]$
    end
        $\hat{e}^k = tr(MP(\bar{d}^k, \epsilon_k))$
        $\hat{d}^k = (\hat{M}\hat{P})^{-1}\hat{e}^k$
    for  $j = 1, \dots, J_{k-1}$
        $\hat{f}_{2j-1}^k = f_{2j-1}^P + \hat{d}_j^k$
        $\hat{f}_{2j}^k = 2\hat{f}_j^{k-1} - \hat{f}_{2j-1}^k$
    end
end
    $M^M \bar{f}^L = \{\bar{f}^0, \hat{e}^1, \dots, \hat{e}^L\}$
```

Proposition 6.1 *Given a discrete sequence $\bar{f}^L$ and a tolerance level ϵ, if the truncation parameters ϵ_k in the modified encoding algorithm (6.3) are chosen so that*

$$\epsilon_k := \epsilon$$

then the sequence $\hat{f}^L = M^{-1}\{\bar{f}^0, \hat{e}^1, \dots, \hat{e}^L\}$ *satisfies*

$$||\bar{f}^L - \hat{f}^L||_p \leq \epsilon \tag{51}$$

for $p = \infty, 1$ *and* 2.

Thus, the modified algorithm for the interpolatory case is stable.

Proof

We apply complete induction over the number of multiresolution's steps at the details. For $n = 1$ we recover the usual error-control (see [31]). Then we suppose the property true for $n = 1, 2, \dots, m-1$, and we will prove it for $n = m$.

Observe that

$$\bar{f}^k_{2j-1} - \hat{f}^k_{2j-1} = \bar{d}^k_j - \hat{d}^k_j$$

$$\bar{f}^k_{2j} - \hat{f}^k_{2j} = \bar{f}^k_j - \hat{f}^k_j$$

Then

$$||\bar{f}^k - \hat{f}^k||_\infty \leq max\{||\bar{d}^k - \hat{d}^k||_\infty, ||\bar{f}^{k-1} - \hat{f}^{k-1}||_\infty\}$$

Since $\hat{f}^0 = \bar{f}^0$, we have (induction hypothesis)

$$||\bar{f}^k - \hat{f}^k||_\infty \leq max\{\epsilon_k, \epsilon_{k-1}, \dots \epsilon_1\} \leq \epsilon$$

Also

$$\begin{aligned} ||\bar{f}^k - \hat{f}^k||_1 &= \frac{1}{J_k}\sum_{j=1}^{J_k} |\bar{f}^k_j - \hat{f}^k_j| \\ &= \frac{1}{J_k}\sum_{i=1}^{J_{k-1}} (|\bar{f}^k_{2j-1} - \hat{f}^k_{2j-1}| + |\bar{f}^k_{2j} - \hat{f}^k_{2j}|) \end{aligned}$$

$$= \frac{1}{J_k}\sum_{i=1}^{J_{k-1}}(|\bar{f}_j^{k-1} - \hat{f}_j^{k-1}| + |\bar{d}_j^k - \hat{d}_j^k|)$$
$$= \frac{1}{2}(||\bar{f}^{k-1} - \hat{f}^{k-1}||_1 + ||\bar{d}^k - \hat{d}^k||_1)$$

thus

$$||\bar{f}^k - \hat{f}^k||_1 = \sum_{k=1}^{L} \frac{1}{2^{L-k+1}}||\bar{d}^k - \hat{d}^k||_1 \leq \epsilon(1 - \frac{1}{2^L}) < \epsilon$$

Similarly

$$||\bar{f}^k - \hat{f}^k||_2^2 = \sum_{k=1}^{L} \frac{1}{2^{L-k+1}}||\bar{d}^k - \hat{d}^k||_2^2 \leq \epsilon^2(1 - \frac{1}{2^L}) < \epsilon^2$$

□

Proposition 6.2 *Given a discrete sequence $\bar{f}^L$ and a tolerance level ϵ, if the truncation parameters ϵ_k in the modified encoding algorithm (6.4) are chosen so that*

$$\epsilon_k := \epsilon \cdot (\frac{1}{2})^{L-k}$$

then the sequence $\hat{f}^L = M^{-1}\{\bar{f}^0, \hat{e}^1, \ldots, \hat{e}^L\}$ satisfies

$$||\bar{f}^L - \hat{f}^L||_p \leq C\epsilon \tag{52}$$

for $p = \infty, 1$ and 2. $C = 2$ for $p = \infty, 1$ and $C = \frac{2}{\sqrt{3}}$ for $p = 2$.

Thus, the modified algorithm for the cell average case is stable.

Proof

We apply complete induction over the number of multiresolution's steps at the details. For $n = 1$ we recover the usual error-control (see [31]). Then we suppose the property true for $n = 1, 2, \ldots, m-1$, and we will prove it for $n = m$.

Observe that

$$\bar{f}_{2j-1}^k - \hat{f}_{2j-1}^k = (\bar{f}_j^{k-1} - \hat{f}_j^{k-1}) + (\bar{d}_j^k - \hat{d}_j^k)$$

$$\bar{f}^k_{2j} - \hat{f}^k_{2j} = (\bar{f}^k_j - \hat{f}^k_j) - (\bar{d}^k_j - \hat{d}^k_j)$$

Then

$$||\bar{f}^L - \hat{f}^L||_\infty \leq ||\bar{f}^0 - \hat{f}^0||_\infty + \sum_{k=1}^{L} ||\bar{d}^k - \hat{d}^k||_\infty$$

Since $\hat{f}^0 = \bar{f}^0$, we have (induction hypothesis)

$$\begin{aligned} ||\bar{f}^L - \hat{f}^L||_\infty &\leq \sum_{k=1}^{L} \epsilon(\frac{1}{2})^{L-k} \\ &= 2\,\epsilon\,(1 - \frac{1}{2^L}) < \ 2\,\epsilon \end{aligned}$$

We obtain the same conclusion for the L^1 norm, since

$$||\bar{f}^L - \hat{f}^L||_1 \leq ||\bar{f}^0 - \hat{f}^0||_1 + \sum_{k=1}^{L} ||\bar{d}^k - \hat{d}^k||_1$$

Finally, from (directly computation)

$$\frac{1}{J_k}\sum_{j=1}^{J_k} |\bar{f}^k_j - \hat{f}^k_j|^2 = \frac{1}{J_{k-1}}\sum_{j=1}^{J_{k-1}} |\bar{f}^{k-1}_j - \hat{f}^{k-1}_j|^2 + \frac{1}{J_{k-1}}\sum_{j=1}^{J_{k-1}} |\bar{d}^k_j - \hat{d}^k_j|^2$$

we obtain

$$\begin{aligned} ||\bar{f}^L - \hat{f}^L||_2^2 &= ||\bar{f}^0 - \hat{f}^0||_2^2 + \sum_{k=1}^{L} ||\bar{d}^k - \hat{d}^k||_2^2 \\ &\leq \epsilon^2 \sum_{k=1}^{L} (\frac{1}{4})^{L-k} = \epsilon^2 \frac{1 - (\frac{1}{4})^L}{1 - \frac{1}{4}} \\ &< \epsilon^2 (1 - \frac{1}{4})^{-1} \end{aligned}$$

□

REMARK 6.3 *Those propositions give us explicit bounds of the error.*

REMARK 6.4 *As we said before, other measure of a sequence is the l^2logl^2 norm:*

$$\mathcal{L}(x) = -\sum_j |x_j|^2 ln|x_j|^2 \tag{53}$$

and the threshold in the details is $\sqrt{\epsilon \cdot exp(\frac{-\mathcal{L}(x)}{||x||^2})}$. The term $exp(\frac{-\mathcal{L}(x)}{||x||^2})$ is directly related to average energy of significant coefficients. With the same ideas we can obtain propositions in this context.

6.3 Numerical Experiments and Conclusions

In this section we present some numerical tests. We will use the notation

I.W.=interpolatory wavelets.

N.I.M.=nonlinear interpolatory multiresolution (ENO non-hierarchical).

I.W.P.=interpolatory wavelets packets.

N.I.M.P.= nonlinear interpolatory multiresolution-packets (ENO non-hierarchical).

We will replace "I" by "C" if we are using cell-averages.

E-C = with error-control.

We will consider two different definitions of compression rate:

Let $D_\epsilon = \{(j,k) : |d_j^k| > \epsilon_k\}$, then we define

$$r_c^1 = \frac{J_L}{|D_\epsilon| + J_0} \tag{54}$$

$$r_c^2 = \frac{(J_L - J_0) - |D_\epsilon|}{(J_L - J_0)} \tag{55}$$

with the second definition we will have full compression when $r_c^2 = 1$.

We will use a fourth order reconstruction and a third order reconstruction in the point-values and in the cell-average frameworks respectively.

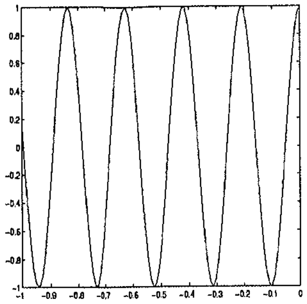

Fig. 10: function f_1

We know the wavelets framework is a good strategy to compress a signal. Nevertheless, in some cases it is necessary to have a better location of the frecuency. The behavior of the following signal can model this problem. We consider $f_1(x) = \cos(30 \cdot x)$ in $[-1, 0]$.

We present in table 6 the improvements of the wavelets packets. When wavelets have frequency problems, the compression using wavelets-packets is improven.

L	I.W	I.W.P
1	$\hat{d}^1 = 65$	$\hat{d}^1 = 65$
2	$\hat{d}^2 + \hat{d}^1 = 65 + 62$	$\hat{d}^2 + \hat{d}^1 = 34 + 62$
3	$\hat{d}^3 + \hat{d}^2 + \hat{d}^1 = 65 + 62 + 32$	$\hat{d}^3 + \hat{d}^2 + \hat{d}^1 = 34 + 35 + 32$

Table 6: Number of details non zeros after truncation, $\epsilon = 5 \cdot 10^{-5}$, $J_L = 256$

The error-control described allow us a control of the error. In table 7 the errors from the error-control algorithms satisfy the theoretical error bounds. Moreover even the linear schemes (without error-control) seem to have stability constants, ($||\bar{f}^L - \hat{f}^L|| \leq C\epsilon$), $C > 1$ in the infinity norm.

On the other hand, non linear techniques are better adapted to the presence of discontinuities [5]. Moreover, ENO reconstruction obtain better resolution in the

I.W.P.	ϵ	r_c^2	$\|\|\cdot\|\|_1$	$\|\|\cdot\|\|_2$	$\|\|\cdot\|\|_\infty$
	1.	1.	$6.33e-01$	$7.05e-01$	$1.00e+00$
	0.1	0.879	$9.14e-03$	$1.50e-02$	$7.46e-02$
	0.01	0.798	$1.57e-03$	$2.77e-03$	$1.09e-02$
	0.001	0.712	$2.38e-04$	$3.48e-04$	$1.39e-03$
I.W.P.(E-C)	ϵ	r_c^2	$\|\|\cdot\|\|_1$	$\|\|\cdot\|\|_2$	$\|\|\cdot\|\|_\infty$
	1.	1.	$6.33e-01$	$7.05e-01$	$1.00e+00$
	0.1	0.879	$9.14e-03$	$1.50e-02$	$7.46e-02$
	0.01	0.794	$1.46e-03$	$2.55e-03$	$9.64e-03$
	0.001	0.708	$2.18e-04$	$3.16e-04$	$9.93e-04$

Table 7: $J_L = 256$, $L = 3$, function f_1

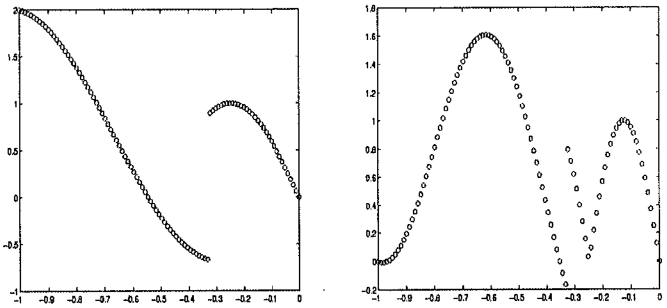

Fig. 11: function f_2, left $\alpha = 3$, $\beta = 2$, right $\alpha = 5$, $\beta = 4$

local extremes than the central reconstruction. In order to see this, we consider the following signal:

$$f_2(j) = \begin{cases} -x_j + \sin(\alpha \cdot \frac{pi}{2} \cdot x_j) & x_j \in [-1, -\frac{1}{3}] \\ |\sin(\beta \cdot \pi \cdot x_j)| & x_j \in (-\frac{1}{3}, 0] \end{cases}$$

where α, β are constants, $x_j = -1 + j \cdot \frac{1}{n}$ and $n = J_L = 256$.

We summarize the conclusions at the tables 8 and 9.

The nonlinear schemes obtain better compression. When L is large we can see the advantage of use the packet framework. In this case the error is smaller than

L	ϵ	I.W	I.W.P	N.I.W	N.I.M.P
1	10^{-3}	3	3	1	1
2	10^{-3}	6	6	2	2
3	10^{-3}	9	9	3	3
4	10^{-3}	14	14	8	8
1	10^{-5}	3	3	1	1
2	10^{-5}	6	6	4	4
3	10^{-5}	36	36	34	34
4	10^{-5}	51	42	49	40

Table 8: Number of details non zeros after truncation, f_2, $\alpha = 3$, $\beta = 2$, $J_L = 256$

L	I.W	I.W.P	N.I.W	N.I.M.P
1	0.976	0.976	0.984	0.984
2	0.969	0.969	0.979	0.979
3	0.839	0.839	0.848	0.848
4	0.787	0.825	0.796	0.833

Table 9: Ratio of compression (55) r_c^2, function f_2, $\alpha = 3$, $\beta = 2$, $J_L = 256$, $\epsilon = 10^{-5}$

the theoretical bounds always, i.e. smaller than ϵ.

Similar conclusions are obtained in the cell-average case table 10. Notice you in the cell average framework the order and the tolerance parameters are smaller than in the point values case (third order and $\epsilon_k = \epsilon(\frac{1}{2})^{L-k}$), then the compression can be smaller too.

Of course, if the function doesn't have frequency problems as

$$f_3(x) = \begin{cases} .5\sin(\pi x) & x \in [0, \frac{2}{3}] \\ -.5\sin(\pi x) & x \in (\frac{2}{3}, 1] \end{cases}$$

the wavelets tools are sufficient. In this case W.P. obtains similar results with

	$Num.of details$	$\|\cdot\|_1$	$\|\cdot\|_2$	$\|\cdot\|_\infty$
I.W.P	15	$1.43e-04$	$2.27e-04$	$7.36e-04$
I.W.P.(E-C)	15	$1.44e-04$	$2.27e-04$	$7.34e-04$
N.I.M.P.	9	$2.21e-04$	$3.45e-04$	$1.06e-03$
N.I.M.P.(E-C)	11	$2.13e-04$	$3.32e-04$	$9.93e-04$
C.W.P	31	$4.27e-05$	$7.47e-05$	$2.90e-04$
C.W.P (E-C)	31	$4.27e-05$	$7.47e-05$	$2.90e-04$
N.C.M.P	25	$2.00e-05$	$5.26e-05$	$3.79e-04$
N.C.M.P (E-C)	25	$2.00e-05$	$5.26e-05$	$3.79e-04$

Table 10: f_2, $\alpha = 5$, $\beta = 4$, $J_3 = 256$, $\epsilon = 10^{-3}$

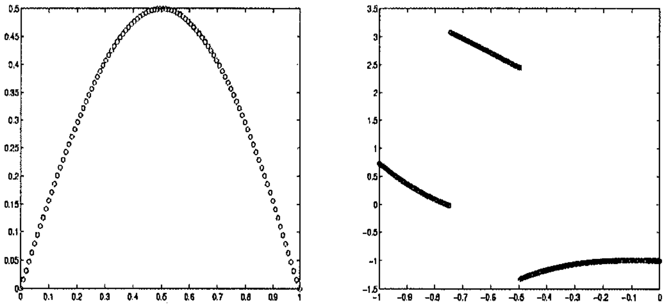

Fig. 12: left function f_3, right function f_4

and without error-control (see table 11).

Now we consider the following signal (Harten's function):

$$f_4(j) = \begin{cases} -\frac{\pi}{2} \cdot x_j^3 + \sin(x_j) & j \in (0, \frac{n}{4}) \\ |\pi \cdot x_j| + \cos(x_j) & j \in (\frac{n}{4}, \frac{n}{2}) \\ 2 \cdot x_j - 1 - 4 \cdot \frac{\sin(\pi \cdot x_j)}{6} & j \in (\frac{n}{2}, n) \end{cases}$$

where $x_j = -1 + j \cdot \frac{1}{n}$ and $n = J_L = 256$.

In figure 13 we can see the advantage to use the nonlinear scheme. With an error less than the tolerance parameter, the nonlinear scheme obtains better compression than the linear one.

	r_c^2	$\|\cdot\|_1$	$\|\cdot\|_2$	$\|\cdot\|_\infty$
I.W.P	0.844	$3.30e-07$	$4.50e-07$	$1.07e-06$
I.W.P.(E-C)	0.844	$3.31e-07$	$4.42e-07$	$1.07e-06$
N.I.M.P.	0.868	$1.58e-06$	$5.84e-06$	$5.76e-05$
N.I.M.P.(E-C)	0.868	$1.58e-06$	$5.84e-06$	$5.77e-05$
C.W.P	0.844	$9.40e-07$	$1.43e-06$	$4.64e-06$
C.W.P (E-C)	0.844	$9.40e-07$	$1.43e-06$	$4.64e-06$
N.C.M.P	0.867	$1.86e-06$	$2.79e-06$	$8.65e-06$
N.C.M.P (E-C)	0.867	$1.86e-06$	$2.79e-06$	$8.65e-06$

Table 11: f_3, $J_3 = 256$, $\epsilon = 10^{-4}$

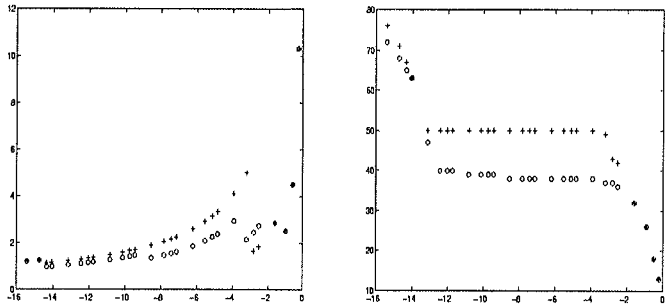

Fig. 13: 'left $log(norm_2)/log(tol)$ versus $log(tol)$ +=linear and o=nonlinear, right Number of non zeros versus $log(tol)$ o=linear and +=nonlinear, tol = tolerance parameter, point-values, $J_L = 256$, $L = 4$, f_4

Finally, in order to put out the profit of the error-control strategy, we consider the function $f_5(x) = 30\cos(100x)$ in $[-1, 0]$. In figure 15 we see as the compression of the E-C is similar than the general case (without error-control). Nevertheless the error is increasing in the general case with the tolerance. The E-C is adapted to obtain an error less than the theoretical bounds, but maintaining a good compression.

Some efficient schemes for representing signals and images are proposed. The

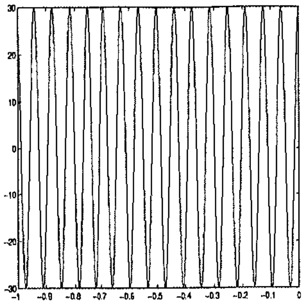

Fig. 14: function f_5

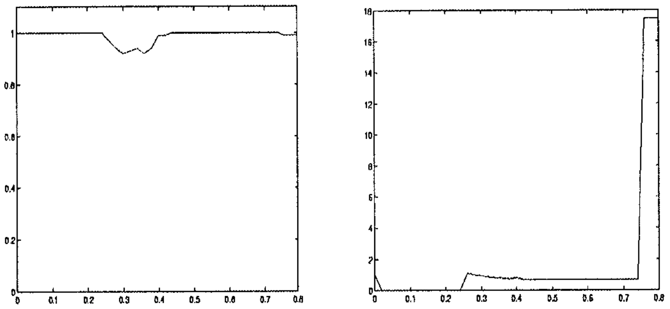

Fig. 15: ‘left $\frac{nnz}{nnz(E-C)}$ versus tolerance=ϵ (nnz=number of nonzeros), right $\frac{||\cdot||_{\infty}}{||\cdot||_{\infty(E-C)}}$ versus tolerance=ϵ, point-values, $J_L = 256$ $L = 4$

advantage of using a multiresolution packets is that it can be adapted to the time-frequency-smoothness characteristic of the underling signal. The wavelet packets transform is very adapted to the time-frequency and this procedure leads to remarkable compression algorithms. Using non-linear reconstruction one can adapted to the presence of discontinuities. So, putting together the wavelet packet's ideas and non-linear reconstructions, like our algorithms, we can have all the mentioned properties and to obtain optimal schemes in this sense.

7 Leyes de Conservación: Una Visión global rápida

Una introducción general a las propiedades de los sistemas de leyes de conservación y sus métodos numéricos asociados puede encontrarse en el libro de LeVeque [46]. Nuestro objetivo en esta sección es destacar los aspectos fundamentales para poder entender las secciones posteriores.

Un sistema físico continuo puede describirse por las leyes de conservación de masa, momento y energía. Es decir, para cada cantidad conservada, la proporción de cambio de la cantidad total en alguna región viene dada por su flujo (convectivo o difusivo) a través de la frontera, más cualquier fuente existente.

Como ya hemos dicho, una cantidad conservada, como la masa, puede ser transportada por flujos convectivos o difusivos. La distinción es que los flujos difusivos están conducidos por gradientes en densidad, mientras los flujos convectivos incluso persisten en ausencia de estos. Nos concentraremos en el caso de transporte convectivo, ignorando difusión (difusión de masa, viscosidad y conductividad térmica) y también términos fuente (como reacciones químicas, excitaciones atómicas, ionizaciones). Hacemos esta simplificación ya que el transporte convectivo requiere tratamiento numérico especializado. Los difusivos y los efectos reactivos pueden ser tratados por métodos numéricos "normales" independientes de aquellos para los términos convectivos. Las leyes de conservación con sólo flujos convectivos son conocidas como "hiperbólicas". Una inmensa serie de fenómenos físicos se modelizan con ayuda de tales sistemas: combustión en motores, astrofísica, plasma,...

7.1 Fenómeno Convectivo, Modelos e Implicaciones Numéricas

Los fenómenos físicos más importantes exhibidos por las leyes de conservación hiperbólicas son *"bulk convection", "waves", discontinuidades de contacto, shocks, y rarefracciones.* Describiremos brevemente los rasgos físicos y modelo matemático para cada efecto.

Los dos primeros fenómenos son simplemente el movimiento de materia de un sitio a otro. La ecuación modelo más simple que describe estos fenómenos es la ecuación lineal convectiva, también es un modelo importante para entender transporte suave en las leyes de conservación. Estos fenómenos se propagan en direcciones bien definidas en contraste con fenómenos como la difusión que se propaga en todas las direcciones.

Los métodos numéricos convenientes para los sistemas hiperbólicos son los upwind que contemplan las diversas direcciones, además de una relación definida entre el espacio y el paso de tiempo. La velocidad de la propagación discreta $\frac{\Delta x}{\Delta t}$ debe ser igual que la velocidad de propagación física. La forma general de esta relación se llama restricción de Courant-Friedrichs-Lewy (CFL), y dice que la velocidad discreta debe ser por lo menos igual de grande que cualquier velocidad característica en el problema.

Una discontinuidad de contacto es un salto persistente, discontinuo en la densidad de masa que se mueve por la transmisión de volumen a través del sistema. Como la difusión de masa es despreciable, el salto persiste. Estos saltos normalmente aparecen en los puntos de contacto de materiales diferentes, por ejemplo, una discontinuidad de contacto puede separar el aceite del agua. Se mueven a velocidad característica y retienen cualquier perturbación que reciben. Así cualquier alteración producida por el método numérico tiende a persistir y aumentar.

Shock: Un shock es un salto espacial en material (presión, temperatura) desarrollado espontáneamente (aunque no necesariamente) de distribuciones suaves y que persiste. Al contrario que la discontinuidad de contacto que debe ponerse en el sistema inicialmente. Los shocks se desarrollan a través de un mecanismo de la regeneración en que los impulsos fuertes se mueven más rápidamente que los débiles. Se pueden modelizar con la ecuación de Burgues:

$$u_t + (\frac{u^2}{2})_x = 0 \tag{56}$$

El movimiento de los shocks no es tan trivial como el de las discontinuidades de contacto y sus velocidades no son evidentes a partir de sus flujos. Es sabido que los métodos numéricos deben estar en forma conservativa para poder "capturar" los shocks.

La aparición expontánea de los shocks tiene dos implicaciones fundamentales para los métodos numéricos. Primero, incluso cuando el dato inicial sea suave los métodos tienen que estar adaptados a gradientes grandes y saltos. Segundo, hay un efecto beneficioso y es que los errores pequeños cerca del shock tienden a disminuirse y desaparecer en el transcurso del tiempo.

Rarefracción: Una rarefracción es un salto o gradiente demasiado grande que se disipa como una expansión suave. Puede modelizarse con la ecuación Burgues más una condición inicial del estilo de $\tanh(\frac{x}{\epsilon})$. Disminuye los errores numéricos y hace más fácil la representación de la solución por polinomios que serán la base de nuestros métodos.

Las consideraciones numéricas a tener en cuenta son:

La condición CFL es necesaria para que el modelo tenga una propagación de información correcta.

Los métodos numéricos deben adaptarse a las direcciones de propagación, así deben de ser "upwind".

En regiones de suavidad es posible obtener alta resolución con interpolación.

Cualquier método para ecuaciones diferenciales puede ser aplicado para la integración temporal, sólo se ha de tener en cuenta la estabilidad.

La forma conservativa es crucial para la captura del shock.

La descomposición característica del sistema se hará de forma local.

7.2 Leyes de Conservación

Los sistemas de leyes de conservación son sistemas de ecuaciones en derivadas parciales (EDP's) tiempo-dependientes. En 1-D se pueden escribir como

$$\frac{\partial}{\partial t}u(x,t) + \frac{\partial}{\partial x}f(u(x,t)) = 0. \tag{57}$$

donde $u : R \times R \longrightarrow R^m$ es un vector m-dimensional de las cantidades conservadas.

Asumiremos que el sistema (57) es hiperbólico. Esto significa que la matriz Jacobiana $f'(u)$ tiene la siguiente propiedad: *Para todo valor de u los valores propios de $f'(u)$ son reales y la matriz es diagonalizable.*

Normalmente el flujo es una función no lineal de u, dando lugar a sistemas no lineales de EDP's. En general, no será posible la obtención de la solución, de ahí la necesidad del estudio de métodos numéricos.

Uno de los sistemas de leyes de conservación con mayor importancia es el de las ecuaciones de Euler de la dinámica de gases. En realidad, las ecuaciones fundamentales de la dinámica de fluidos son las de Navier-Stokes, pero estas ecuaciones incluyen los efectos de la viscosidad, y así los flujos no sólo dependen de las variables de estado sino también de los gradientes de éstas. En consecuencia estas ecuaciones dejan de ser hiperbólicas. Por otro lado, si las sustancias están lo suficientemente diluidas, como puede ser el caso de un gas, los efectos de la viscosidad pueden ser despreciados, apareciendo en este caso las ecuaciones de Euler. En una dimensión estas ecuaciones se escriben:

$$\frac{\partial}{\partial t}\begin{pmatrix} \rho \\ \rho v \\ E \end{pmatrix} + \frac{\partial}{\partial x}\begin{pmatrix} \rho v \\ \rho v^2 + P \\ v(E+P) \end{pmatrix} = 0$$

donde $\rho = \rho(x,t)$ denota la densidad, v la velocidad, ρv el momento, E la energía, y P la presión. La presión P vendrá dada por la ecuación de estado.

Consideremos ahora la ecuación escalar no lineal

$$u_t + f(u)_x = 0 \tag{58}$$

donde $f(u)$ es una función no lineal de u.

Una forma natural de obtener una definición de solución que no use la diferenciabilidad es regresar a la forma integral de la ecuación. Sea $\phi \in C_0^1(R \times R)$ una función test. Si multiplicamos $u_t + f_x = 0$ por $\phi(x,t)$ e integramos respecto al tiempo y al espacio obtendremos:

$$\int_0^\infty \int_{-\infty}^\infty \phi u_t + \phi f(u)_x dxdt = 0. \tag{59}$$

Integrando por partes,

$$\int_0^\infty \int_{-\infty}^\infty \phi_t u + \phi_x f(u) dxdt = -\int_{-\infty}^\infty \phi(x,0)u(x,0)dx \tag{60}$$

Definición 6 *Una función $u(x,t)$ es una solución débil de la ley de conservación (58) si se satisface (60) para toda función $\phi \in C_0^1(R \times R)$.*

Existen situaciones en las que la solución débil no es única y son necesarias otras condiciones para escoger la solución físicamente relevante.

Definición 7 *Se dice que $u(x,t)$ es la solución de entropía si existe una constante $E > 0$ tal que para todo $a > 0$, $t > 0$ y $x \in R$,*

$$\frac{u(x+a,t)-u(x,t)}{a} < \frac{E}{t} \tag{61}$$

Oleinik [46]

Nota: Existen versiones más simples de esta condición para distintos casos particulares.

8 Local piecewise polynomial reconstruction of numerical fluxes for nonlinear scalar conservation laws

In this section a local (at least third order accurate) shock capturing method for hyperbolic conservation laws is presented. An extension to fifth order accurate is presented also. To complete the scheme, we use a special family of Runge-Kutta time integration schemes introduced by Shu and Osher that have a "Total Variation Diminishing" (TVD) property.

Essential Non-Oscillatory (ENO) methods, constructed by Harten, Osher, Engquist, and Chakravarthy [37], [38], [39], are a class of high accuracy shock capturing numerical methods for hyperbolic systems of conservation laws, based on upwind biased differencing in local characteristic fields. With these methods have achieved excellent results in a great variety of compressible flow problems. The most efficient implementation of ENO methods has been investigated by Shu and Osher [51]-[52], where they reconstructed numerical fluxes from point values. Originally, ENO schemes were based on the reconstruction of the solution from cell average. Marquina [48] introduced a new local third order accurate shock capturing method (PHM piecewise hyperbolic method). To design this method a new concept of local smoothing is introduced to prevent the increasing of total variation of the solution near discontinuities and to achieve third order accuracy. The main advantage of this method lies on the property that it is localer than ENO and TVD upwind schemes of the same order, (and, thus, giving better resolution of corners), because numerical fluxes depend only on four variables. This method becomes efficient since it is low cost and it is not sensitive to the Courant-Friedrichs-Lewy (CFL) number. Our (third order) method is quiet similar to PHM method, but it is based on a simpler reconstruction and, thus it has lower cost and can be easily tractable. It becomes third order accurate in smooth regions of the solution, except at local extrema where it may degenerate, but giving (at least) the same accuracy than PHM methods and better than TVD methods. We will introduce an extension to fifth order accuracy. In order to achieve fifth order accuracy, we will have to modify the original local smoothing introduced in the PHM method.

The primary goal of all these schemes is to develop a general purpose numerical method for systems of conservation laws that has high accuracy (at least third order) in smooth regions and captures the motion of unresolved steep gradients spurious oscillations.

To complete the schemes, Shu and Osher developed a special family of Runge-Kutta time integration schemes that are easy to implement, have good stability properties and have a TVD property see [51]. The TVD property prevents the time stepping scheme from introducing spurious spatial oscillations into upwind-biased spatial discretization.

In this section, we consider numerical approximations to weak solutions of systems of hyperbolic conservation laws of the type:

$$u_t + \sum_{i=1}^{d} f_i(u)_{x_i} = 0 \tag{62}$$

$$u(x,0) = u_0(x), \tag{63}$$

the initial data $u_0(x)$ are supposed to be piecewise smooth functions that either periodic or of compact support.

We begin with the 1-D scalar nonlinear problem:

$$u_t + f(u)_x = 0 \tag{64}$$

$$u(x,0) = u_0(x), \tag{65}$$

Let be $u_j^n = u_h(x_j, t_n)$ denotes a numerical approximation to the exact solution $u(x_j, t_n)$ of (64)-(65) defined on a computational grid $x_j = jh$, $t_n = n\Delta t$ in conservation form:

$$u_j^{n+1} = u_j^n - \lambda(\hat{f}_{j+\frac{1}{2}}^n - \hat{f}_{j-\frac{1}{2}}^n) \tag{66}$$

where $\lambda = \frac{\Delta t}{h}$ and the numerical flux is a function of 2k variables

$$\hat{f}_{j+\frac{1}{2}}^n = \hat{f}(u_{j-k+1}^n, \ldots, u_{j+k}^n) \tag{67}$$

which is consistent with (64), i.e.

$$\hat{f}(u, \ldots, u) = f(u) \tag{68}$$

The importance of the following lemma (see [51]) is because it implies that approximating the numerical flux $\hat{f}_{j+\frac{1}{2}}$ to a high order accuracy it is enough to reconstruct $g(x_{j+\frac{1}{2}})$ (see equation (69)) up to the same order.

Lemma 8.1 (Shu and Osher) *If a function $g(x)$ satisfies*

$$f(u(x)) = \frac{1}{h} \int_{x-\frac{h}{2}}^{x+\frac{h}{2}} g(\xi) d\xi \tag{69}$$

then

$$f(u(x))_x = \frac{g(x + \frac{h}{2}) - g(x - \frac{h}{2})}{h}$$

We will present a local third order accurate method by using a piecewise reconstruction of the function g. The obtained method is localer in the sense that the numerical flux depends on less points than the corresponding ENO and TVD schemes.

The one time step procedure described in this paper is assumed to be total variation stable for scalar 1-D nonlinear problems under suitable CFL restriction

$$\lambda = \frac{\Delta t}{h} \leq \lambda_0$$

and λ_0 is inversely proportional to $max|f'(u)|$ as usual.

We will check our methods are "Local Total Variation Bounded" (LTVB), as ENO and PHM methods. A maximum principle appears to be necessary to prove the TVB property of the schemes.

The section is organized as follows: section 8.1 contains the reconstruction step, in next section the complete algorithm is presented, in section 8.4 we generalized the method to fifth order, in section 8.5 we extend the implementation to nonlinear systems and to multi-dimensions space, and finally, in section 8.6 some numerical experiments are studied.

8.1 Piecewise reconstruction

The most important step in our method, as well as in the above methods, is the reconstruction step. We need to have a reconstruction of the function g, at least, up to third order.

Let $g(x)$ be a piecewise smooth function that is either periodic or of compact support. We have defined a computational grid $x_j = jh$, j integer, $h > 0$, where the cells are

$$C_j = \{x : x_{j-\frac{1}{2}} \leq x \leq x_{j+\frac{1}{2}}\} \tag{70}$$

where $x_{j+\frac{1}{2}} = x_j + \frac{1}{2}h$.

Our grid data are:

(i) For every j the mean value of $g(x)$ in C_j, v_j is given, i.e.:

$$v_j = \frac{1}{h}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} g(\xi)d\xi, \tag{71}$$

(ii) For every j, $d_{j+\frac{1}{2}}$ are given:

$$d_{j+\frac{1}{2}} = \frac{v_{j+1} - v_j}{h}, \tag{72}$$

$(d_{j+\frac{1}{2}} = g^{'}(x_{j+\frac{1}{2}}) + O(h^2))$.

Let $r_j(x)$ a reconstruction of $g(x)$ in C_j up to third order accuracy, i.e., that every time $g(x)$ is smooth enough at x in C_j, then

$$g(x) - r_j(x) = O(h^3).$$

To get consistent third order accurate reconstruction procedures, we required the following conditions for every j:

$$v_j = \frac{1}{h}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} r_j(\xi)d\xi, \tag{73}$$

$$d_{j+\frac{1}{2}} = r_j^{'}(x_{j+\frac{1}{2}}). \tag{74}$$

Taylor series expansions show that conditions (73) and (74) imply third order accuracy. To get local total variation bounded methods of reconstruction, in general, it would be necessary to correct the values of $d_{j+\frac{1}{2}}$.

Next, we look at local piecewise polynomic reconstructions. For our purpose we need the following lemma.

Lemma 8.2 *We consider the generic cell*

$$C_0 = \{x : |x - x_0| \leq \frac{h}{2}\}. \tag{75}$$

Let θ_1 and θ_2 be real numbers between -1 and 1, $\theta_1 \neq \theta_2$. We set $x(\theta_1) = x_0 + \theta_1 \frac{h}{2}$ and $x(\theta_2) = x_0 + \theta_2 \frac{h}{2}$. Let v_0 be the mean value of $g(x)$ in C_0. Let d_1, d_2 be real numbers. Then the following polynomial:

$$p_0(x) = a_0 + a_1 x + a_2 x^2, \tag{76}$$

where:

$$a_2 = \frac{1}{2}(\frac{d_2 - d_1}{x(\theta_2) - x(\theta_1)}),$$

$$a_1 = d_1 - x(\theta_2)\frac{d_2 - d_1}{x(\theta_2) - x(\theta_1)},$$

$$a_0 = v_0 - \frac{h^2}{12}a_2,$$

is such that:

$$v_0 = \frac{1}{h}\int_{x_0-\frac{h}{2}}^{x_0+\frac{h}{2}} p_0(\xi)d\xi, \tag{77}$$

$$p_0^{'}(\theta_1) = d_1, \tag{78}$$

$$p_0^{'}(\theta_2) = d_2. \tag{79}$$

Moreover, the derivative of $p_0(x)$ *at the midpoint* $x(\frac{\theta_1+\theta_2}{2})$ *is*

$$p_0^{'}(x(\frac{\theta_1+\theta_2}{2})) = \frac{d_1+d_2}{2}. \tag{80}$$

Proof

The proof is straightforward.

□

Since our reconstruction is local, we restrict our discussion to the cell C_0 and the grid data of the cell: $v_0, d_{-\frac{1}{2}}$, and $d_{\frac{1}{2}}$. To fit the reconstruction, we establish formulas to obtain the coefficients with $\theta_1 = 0$. If $d_{-\frac{1}{2}} \cdot d_{\frac{1}{2}} > 0$, the value assigned to d_1 is the average (80). If $d_{-\frac{1}{2}} \cdot d_{\frac{1}{2}} \leq 0$ (transition cell), we change the derivative with largest absolute value by other one multiplied by h^2, thus, the reconstruction on this cell may degenerate to second order. In spite of the smoothing made on transition cells, the method of reconstruction is not LTVB. For nonlinear fluxes, the method becomes "unstable". As PHM scheme, we will assign a different value to the central derivative d_1. If C_0 is a nontransition cell, then we define d_1 by

$$d_1 = \frac{2d_{-\frac{1}{2}} \cdot d_{\frac{1}{2}}}{d_{-\frac{1}{2}} + d_{\frac{1}{2}}}, \tag{81}$$

which is the harmonic mean of $d_{-\frac{1}{2}}$ and $d_{\frac{1}{2}}$ (smaller in absolute value than (80)). Then the algorithm defines the polynomial such that its derivative interpolates d_1 at x_0 and the lateral grid derivative with smallest absolute value. Taylor series expansion arguments show that the preprocessing defined in this way is consistent, since the harmonic mean provides an $O(h^2)$ approximation of the derivative at x_0. Transition cells are treated analogously to the first algorithm, but use the harmonic mean instead of (80).

Theorem 8.1 *The above method of reconstruction of the function $g(x)$ is a local preprocessed polynomic reconstruction procedure that is LTVB.*

Proof

Let us choose a number $h > 0$, such that there are at least two cells between two jumps of $g(x)$. Since $g(x)$ is a piecewise smooth function, it is easy to see that there exists a constant $M > 0$ depending only on derivatives of g in smooth regions, such that for all j, except for a finite number of "isolate" j's (for which $d_{j+\frac{1}{2}} = O(h^{-1})$), $|d_{j+\frac{1}{2}}| < M$.

If C_j is a nontransition cell and $|d_{j-\frac{1}{2}}| \leq |d_{j+\frac{1}{2}}|$, then the preprocessed derivatives at the endpoint of the cells are the following:

$$dl_j = d_{j-\frac{1}{2}}, \tag{82}$$

$$|dr_j| = |a_1 + 2 \cdot a_2 \frac{h}{2}| \leq |2 \cdot dl_j| + |\frac{dl_j}{h}|h \leq 3 \cdot dl_j \leq 3 \cdot M \tag{83}$$

(for PHM method we obtain $|dr_j| \leq 4 \cdot M$)

Thus

$$TV(p_j) \leq h\frac{4 \cdot M}{2} = 2 \cdot Mh. \tag{84}$$

The argument is similar for transition cells.

□

REMARK 8.1 *With the same notation, PHM method is based in the hyperbola,*

$$h_0 = v_0 + d_1 \cdot h \cdot \frac{1}{\alpha^2} \cdot (log(\frac{2-\alpha(1-\theta_1)}{2+\alpha(1+\theta_1)}) - \frac{h}{(x-x_0) - \frac{h}{2} \cdot (\theta_1 + \frac{2}{\alpha})}) \quad .$$

where $\alpha = \frac{2}{\theta_2 - \theta_1} \cdot (1 - \sqrt{\frac{d_1}{d_2}})$.

Special treatment when the derivatives are smaller than $O(h^2)$ is considered.

On the other hand, the following algorithm determines the lateral numerical derivatives in the ENO third order method on the cell C_j,

```
if $|d_{j-\frac{1}{2}}| \leq |d_{j+\frac{1}{2}}|$ then
  $dl_j = d_{j-\frac{1}{2}}$
  if $|d_{j+\frac{1}{2}} - d_{j-\frac{1}{2}}| \leq |d_{(j-1)+\frac{1}{2}} - d_{(j-1)-\frac{1}{2}}|$ then
    $dr_j = d_{j+\frac{1}{2}}$
  else
    $dr_j = dl_j + (d_{(j-1)+\frac{1}{2}} - d_{(j-1)-\frac{1}{2}})$
else
  $dr_j = d_{j+\frac{1}{2}}$
  if $|d_{j+\frac{1}{2}} - d_{j-\frac{1}{2}}| \leq |d_{(j+1)+\frac{1}{2}} - d_{(j+1)-\frac{1}{2}}|$ then
    $dl_j = d_{j-\frac{1}{2}}$
  else
    $dl_j = dr_j - (d_{(j+1)+\frac{1}{2}} - d_{(j+1)-\frac{1}{2}})$
```

8.2 Piecewise Methods

In this section, we introduce the final algorithm. It is based on the first order Roe scheme, with the entropy-fix correction due to Shu and Osher [52], for local piecewise reconstruction. We restrict our description of the algorithm to the computation of numerical fluxes $\hat{f}_{j+\frac{1}{2}}$.

The numerical fluxes are reconstructed from the upwind side, except that if the wind changes direction at the cell, then a local Lax-Friedrichs flux decomposition is performed. The upwind side is determined according to the local sign of $f'(u)$ at $x_{j+\frac{1}{2}}$. In this case we have the Roe speed

$$\bar{a}_{j+\frac{1}{2}} = \frac{f(u^n_{j+1}) - f(u^n_j)}{u^n_{j+1} - u^n_j} \tag{85}$$

to determine the sign of $f'(x_{j+\frac{1}{2}})$.

ALGORITHM

STEP 1: Computation of grid data.

From u^n_j we compute the grid data by means of:

$$v_j = f(u^n_j) \tag{86}$$

$$d_{j+\frac{1}{2}} = \frac{v_{j+1} - v_j}{h} \tag{87}$$

for all j.

STEP 2: Computation of the reconstruction for every j.

STEP 3: For every j **do**

If $f'(u)$ does not changes of sign between u^n_j and u^n_{j+1}, then

Upwindness Phase

$$\bar{a}_{j+\frac{1}{2}} = \frac{v_{j+1} - v_j}{u_{j+1} - u_j} \tag{88}$$

(Roe-speed)

If $\bar{a}_{j+\frac{1}{2}} \geq 0$ then

$$\hat{f}_{j+\frac{1}{2}} = p(v_j, d_{j-\frac{1}{2}}, d_{j+\frac{1}{2}}, r) \tag{89}$$

else

$$\hat{f}_{j+\frac{1}{2}} = p(v_{j+1}, d_{(j+1)-\frac{1}{2}}, d_{(j+1)+\frac{1}{2}}, l) \tag{90}$$

else

Flux decomposition phase

$M_{j+\frac{1}{2}} = max_{u_j^n < u < u_{j+1}^n} |f'(u)|$.

$v_k^+ = \frac{1}{2}(v_k + M_{j+\frac{1}{2}} u_k^n)$ $k = j-1, j, j+1$.

Computation of f^+, as (89).

$v_k^- = \frac{1}{2}(v_k - M_{j+\frac{1}{2}} u_k^n)$ $k = j, j+1, j+2$

Computation of f^-, as (90).

$\hat{f}_{j+\frac{1}{2}} = f^+ + f^-$.

Remark 8.2 *$p(v_j, d_1, d_2, l - r)$ is the reconstruction of the flux (polynomial of degree two) evaluate at the left or at the right of the cell C_j respectively.*

We will refer this method as PPHM (piecewise polynomic harmonic method).

8.3 A compararison of the methods

In smooth regions both methods (PHM and PPHM) are third order accurate, thus their difference is of size $O(h^3)$. Now, we analyze what happens in presence of singularities.

Let us choose a number $h > 0$, such that there are at least two cells between two jumps of $g(x)$. We assume $d_{j-\frac{1}{2}} = O(1)$ and $d_{j+\frac{1}{2}} = O(\frac{1}{h})$ (the other case is similar).

Let p_j and h_j the polynomic and hyperbolic reconstruction respectively.

It is easy to check,

$$p_j(x_{j+\frac{1}{2}}) = v_0 + 1.16 \cdot d_{j-\frac{1}{2}} h, \tag{91}$$

$$h_j(x_{j+\frac{1}{2}}) = v_0 + 1.55 \cdot d_{j-\frac{1}{2}} h. \tag{92}$$

Thus, the difference is of $O(h)$ in regions with a singularity.

REMARK 8.3 *If we consider the mean of the hyperbola instead the harmonic mean, we obtain*

$$p_j(x_{j+\frac{1}{2}}) = v_0 + 2.5 \cdot d_{j-\frac{1}{2}} h. \tag{93}$$

REMARK 8.4 *In a transition cell we can use first a translation, then the reconstruction and finally the inverse translation. In some numerical experiments we will analize this technique (we will refer this reconstruction by PPHM*).*

8.4 Fifth order reconstruction

The main advantage of PPHM method from PHM method is the use of a polynomic reconstruction. In this section, we present a local fifth order accurate shock capturing method, as a generalization of the above third order method.

Now, our grid data are:

(i) For every j the mean value of $g(x)$ in C_j, v_j is given, i.e.:

$$v_j = \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} g(\xi) d\xi, \tag{94}$$

(ii) For every j, $d_{j+\frac{1}{2}}$ and $c_{j+\frac{1}{2}}$ are given:

$$d_{j+\frac{1}{2}} = \frac{v_{j+1} - v_j}{h} - \frac{1}{24h}(v_{j-1} - 3v_j + 3v_{j+1} - v_{j+2}), \tag{95}$$

$$c_{j+\frac{1}{2}} = \frac{v_{j-1} - 3v_j + 3v_{j+1} - v_{j+2}}{h^3}, \tag{96}$$

$(d_{j+\frac{1}{2}} = g'(x_{j+\frac{1}{2}}) + O(h^4))$ and $c_{j+\frac{1}{2}} = g'''(x_{j+\frac{1}{2}}) + O(h^2))$.

To get consistent fifth order accurate reconstruction procedures, we required the following conditions for every j:

$$v_j = \frac{1}{h}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} r_j(\xi)d\xi, \tag{97}$$

$$d_{j+\frac{1}{2}} = r_j'(x_{j+\frac{1}{2}}), \tag{98}$$

$$c_{j+\frac{1}{2}} = r_j'''(x_{j+\frac{1}{2}}). \tag{99}$$

Taylor series expansions show that conditions (97)-(98) imply fifth order accuracy. We have a similar lemma:

Lemma 8.3 *We consider the generic cell*

$$C_0 = \{x : |x - x_0| \leq \frac{h}{2}\}. \tag{100}$$

Let θ_1 and θ_2 be real numbers between -1 and 1 $(\theta_1 \neq \theta_2)$. We set $x(\theta_1) = x_0 + \theta_1 \frac{h}{2}$ and $x(\theta_2) = x_0 + \theta_2 \frac{h}{2}$. Let v_0 be the mean value of $g(x)$ in C_0. Let d_1, d_2, c_1, c_2 be real numbers. Then the following polynomial:

$$p_0(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 \tag{101}$$

where

$$a_4 = \frac{1}{24}\left(\frac{c_2 - c_1}{x(\theta_2) - x(\theta_1)}\right)$$

$$a_3 = \frac{1}{6}(c_2 - x(\theta_2)\frac{c_2 - c_1}{x(\theta_2) - x(\theta_1)})$$

$$a_2 = \frac{1}{2}(\frac{A_1 - A_2}{x(\theta_2) - x(\theta_1)})$$

$$a_1 = A_1 - x(\theta_2)\frac{A_1 - A_2}{x(\theta_2) - x(\theta_1)}$$

$$a_0 = v_0 - \frac{h^2}{12}a_2 - \frac{h^4}{80}a_4$$

$A_1 = d_2 - 3a_3x(\theta_2)^2 - 4a_4x(\theta_2)^3$ *and* $A_2 = d_1 - 3a_3x(\theta_1)^2 - 4a_4x(\theta_1)^3$ *is such that:*

$$v_0 = \frac{1}{h}\int_{x_0-\frac{h}{2}}^{x_0+\frac{h}{2}} p_0(\xi)d\xi. \quad (102)$$

$$p'(\theta_2) = d_2. \quad (103)$$

$$p'(\theta_1) = d_1. \quad (104)$$

$$p'''(\theta_2) = c_2. \quad (105)$$

$$p'''(\theta_1) = c_1. \quad (106)$$

Moreover,

$$p_0'(x(\frac{\theta_1+\theta_2}{2})) = \frac{d_1+d_2}{2} - \frac{h^2}{8}\frac{c_1+c_2}{2}, \quad (107)$$

$$p_0'''(x(\frac{\theta_1+\theta_2}{2})) = \frac{c_1+c_2}{2}. \quad (108)$$

To obtain stability we have to modified the original polynomial. We use a similar idea than third order methods. The most important drawback is to maintain the order, because the harmonic mean provides an $O(h^2)$ approximation of the derivative at the midpoint only.

Let $p_0'(x) = A + Bx + C\frac{x^2}{2} + E\frac{x^3}{6}$ the first derivative of the polynomic reconstruction. We will obtain the coefficients using controled values only and thus we will have the LTVB property (p is a polynomial).

Notice that

$$\hat{d} := \frac{2\cdot d_{j+\frac{1}{2}}d_{j-\frac{1}{2}}}{d_{j+\frac{1}{2}}+d_{j-\frac{1}{2}}} = \frac{d_{j+\frac{1}{2}}+d_{j-\frac{1}{2}}}{2} - \frac{(B\frac{h}{2})^2}{\frac{d_{j+\frac{1}{2}}+d_{j-\frac{1}{2}}}{2}}O(h^4) \tag{109}$$

Applying the interpolatory conditions we obtain

$$d_{j+\frac{1}{2}} + O(h^4) = A + B\frac{h}{2} + C\frac{\frac{h}{2}^2}{2} + E\frac{\frac{h}{2}^3}{6}, \tag{110}$$

$$d_{j-\frac{1}{2}} + O(h^4) = A - B\frac{h}{2} + C\frac{\frac{h}{2}^2}{2} - E\frac{\frac{h}{2}^3}{6}, \tag{111}$$

The coefficients C, E can be evaluate using the harmonic mean. Now, we will obtain A and B. Moreover, $B\frac{h}{2} = d_{j+\frac{1}{2}} - A - C\frac{(\frac{h}{2})^2}{2} - E\frac{(\frac{h}{2})^3}{6}$ (therefore we need to know A only).

Assume that $|d_{j+\frac{1}{2}}| \leq |d_{j-\frac{1}{2}}|$ (the other case is symmetric).

Let S such that $A = p_0'(0) = \frac{2\cdot d_{j+\frac{1}{2}}d_{j-\frac{1}{2}}}{d_{j+\frac{1}{2}}+d_{j-\frac{1}{2}}} + S\cdot(\frac{h}{2})^2 + O(h^4)$.

On the other hand,

$$\begin{aligned} d_{j-\frac{1}{2}} + O(h^4) &= 2A - d_{j+\frac{1}{2}} + C(\frac{h}{2})^2 \\ &= 2\hat{d} + 2S(\frac{h}{2})^2 - d_{j+\frac{1}{2}} + C(\frac{h}{2})^2 \\ &= d_{j-\frac{1}{2}} - 2\frac{B^2(\frac{h}{2})^2}{\frac{d_{j+\frac{1}{2}}+d_{j-\frac{1}{2}}}{2}} + 2S(\frac{h}{2})^2 + (\frac{h}{2})^2 C \\ &= d_{j-\frac{1}{2}} + 2h^2(S + \frac{C}{8} - \frac{\frac{B^2}{4}}{\frac{d_{j+\frac{1}{2}}+d_{j-\frac{1}{2}}}{2}}) \end{aligned} \tag{112}$$

Besides,

$$B = \frac{d_{j+\frac{1}{2}} - \hat{d}}{\frac{h}{2}} - 2Sh - (\frac{h}{2})\frac{C}{2} - (\frac{h}{2})^2\frac{E}{6} \tag{113}$$

Let $T := 2S - \frac{C}{4}$ and $\hat{d}_0 := \frac{d_{j+\frac{1}{2}} - \hat{d}}{\frac{h}{2}}$, then $B^2 = (\hat{d}_0)^2 - 2Th\hat{d}_0 + O(h^2)$.

From (113) we obtain,

$$S + \frac{C}{8} - (\frac{\frac{B^2}{4}}{\frac{d_{j+\frac{1}{2}}+d_{j-\frac{1}{2}}}{2}}) = O(h^2) \tag{114}$$

Thus,

$$\frac{T}{2} - \frac{\frac{((\hat{d}_0)^2 - 2Th\hat{d}_0)}{4}}{\frac{d_{j+\frac{1}{2}}+d_{j-\frac{1}{2}}}{2}} = O(h^2)$$

$$2T(\frac{\hat{d}}{4} + h\frac{\hat{d}_0}{4}) = \frac{(\hat{d}_0)^2}{4} + O(h^2)$$

Therefore

$$\begin{aligned} T &= \frac{(\hat{d}_0)^2}{2(\hat{d} + h\hat{d}_0)} + O(h^2) \\ &= \frac{(\hat{d}_0)^2}{2(\hat{d} + h\hat{d}_0)} + O(h^2) \\ &= \frac{(\hat{d}_0)^2}{2\hat{d}}(1 - h\frac{\hat{d}_0}{\hat{d}}) + O(h^2) \end{aligned}$$

Moreover,

$$\begin{aligned} |h\frac{\hat{d}_0}{\hat{d}}| &= |2\frac{d_{j+\frac{1}{2}}}{\hat{d}} - | \\ &= 2(1 - \frac{d_{j+\frac{1}{2}}}{\hat{d}}) \\ &< 2(1 - \frac{d_{j+\frac{1}{2}}}{2d_{j+\frac{1}{2}}}) \\ &= 1 \end{aligned}$$

Thus, we have controled T and therefore A.

Theorem 8.2 *The above fifth order method of reconstruction of the function $g(x)$ is a local preprocessed polynomic reconstruction procedure that is LTVB.*

REMARK 8.5 *With the same ideas, we can obtain controled expressions for the grid data.*

REMARK 8.6 *Other fifth order accurate scheme is WENO-5 (see [47]). It is based in an special nonlinear combination of the three polynomials of ENO-3 scheme.*

REMARK 8.7 *There are no convenient fourth order or higher TVD Runge-Kutta methods; they do exist, but they only maintain the TVD property when used with special, more complicated spatial discretizations. The standard fourth order accurate Runge-Kutta method can be used, but it is not TVD. This means it could cause spurious spatial oscillations, though in practice this has not been a problem. The third order TVD method is generally recommended, since it has the greatest accuracy and largest time step stability region of time TVD schemes. Due to its large stability region (which includes a segment of purely imaginary linear growth rates), for a sufficiently small time step it is guaranteed to be linearly stable for the entire class of problems considered here.*

8.5 Implementations in Multi-Dimensions and Systems

A special advantage of this type of methods [51], is their relative simplicity in multidimensions. The scalar algorithm is applied to each of the terms $f_i(u)_{x_i}$ in (62), keeping all other variables fixed. A typical CFL restriction $\frac{\Delta t}{\Delta x} max_u |f'(u)| \leq \lambda_0$ will be replaced by $\Delta t max_u \sum_{i=1}^{d} (\frac{1}{\Delta x_i}) |f_i'(u)| \leq \lambda_0$.

For nonlinear systems, we simply apply the scalar algorithms in each (local) characteristic field. For instance in 1-D, let $A_{j+\frac{1}{2}}$ be some "average" Jacobian at

$x_{j+\frac{1}{2}}$ (see remark 8.8). We denote the eigenvalues and left and right eigenvectors of $A_{j+\frac{1}{2}}$ by $\lambda^{(p)}_{j+\frac{1}{2}}$, $l^{(p)}_{j+\frac{1}{2}}, r^{(p)}_{j+\frac{1}{2}}$, $p = 1, \ldots, m$, normalized so that

$$l^{(p)}_{j+\frac{1}{2}} \cdot r^{(p)}_{j+\frac{1}{2}} = \delta_{pq}$$

For any vector $\mathbf{a}$,

$$a^{(p)} = l^{(p)}_{j+\frac{1}{2}} \cdot \mathbf{a} \tag{115}$$

is the component of $\mathbf{a}$ in the pth (local) characteristic field, because

$$\mathbf{a} = \sum_{p=1}^{m} a^{(p)} r^{(p)}_{j+\frac{1}{2}} \tag{116}$$

The algorithm now becomes: same step 1 changing vectors to bold face letters. Computation of the numerical flux $\hat{f}^{(p)}_{j+\frac{1}{2}}$ using

$$\bar{a}_{j+\frac{1}{2}} = \lambda^{(p)}_{j+\frac{1}{2}}$$

and

$$M^{(p)}_{j+\frac{1}{2}} = max_{u \in L(u_j, u_{j+1})} |\lambda^{(p)}(u)|$$

where $L(u_j, u_{j+1})$ is some curve in phase space connecting u_j and u_{j+1}. Use (116) to get $\hat{f}_{j+\frac{1}{2}}$.

In the Euler equations of gas dynamics,

$$M^{(p)}_{j+\frac{1}{2}} = max(|\lambda^{(p)}(u_j), |\lambda^{(p)}(u_{j+1})).$$

REMARK 8.8 *The Jacobian matrix of the convective flux vector is quite important to any characteristic based scheme, since it defines the local linearization of the nonlinear problem. It determines the transformation to the local characteristic fields, and thus what the upwind directions are as well as what quantities are to be upwind differenced.*

To evaluate these linearization we can use linear average, or, in the case of Euler equations of gas dynamics the Roe average. But, when the states differ greatly across the cell wall, using such an intermediate state in the transformation adds subtle spurious features to the solution. As an alternative, Donat and Marquina [28] recommend obtaining the wall flux from a splitting procedure based on fluxes computed separately from the left and right sides, but this topic is beyond the scope of this thesis.

8.6 Numerical Results and Concluding remarks

We have run most examples for different CFLs and time levels, but here we only include what we consider as representatives. In all the experiments h=(b-a)/n and m=number of time step. If we are considering the Euler's equation we will plot density only.

In figures 16-17 we consider Burgues equation ($f(u) = \frac{u^2}{2}$) in 1-D in $(-1.25, 1.25)$, with initial data:

$$u(x,0) = \begin{cases} 1 & |x| \leq 0.5 \\ 0 & else \end{cases}$$

We can see that both methods (PHM and PPHM) retain accuracy in spite of the presence of jumps in the derivative of the solution, which shows the good resolution of corners in both methods.

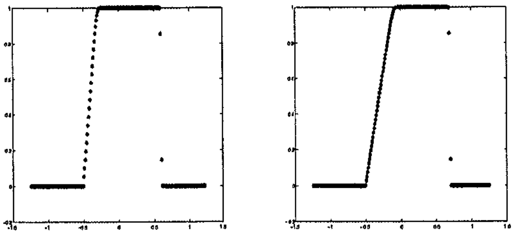

Fig. 16: *=PHM,o=PPHM,left n=256,m=100,right n=256,m=200,CFL=0.2

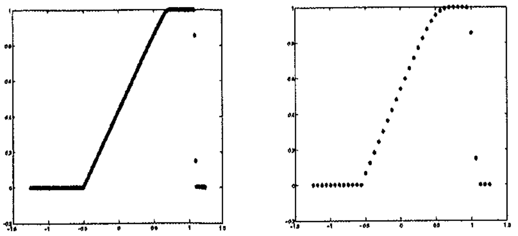

Fig. 17: *=PHM,o=PPHM,left n=256,m=600,right n=40,m=80,CFL=0.2

In figure 18 there is an example of non-convex flux. We observe the considerable improvements in shock transition.

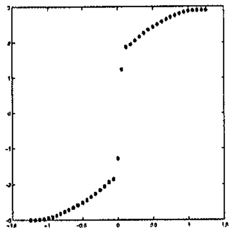

Fig. 18: non-convex flux,PPHM

In figures 19-20 we examine Burgues equation with data $2 + \sin(4 \cdot \pi \ x)$ in $(-1, 1)$. In this case both algorithms are identical.

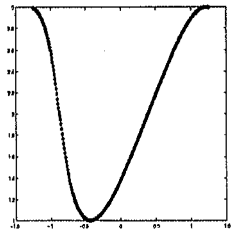

Fig. 19: *=PHM, o=PPHM, n=256, m=100, CFL=0.2

In figure 21 we consider in $(-10, 10)$ the initial data (Euler's equations):

$$u(x,0) = \begin{cases} (1,0,2.5) & -0.5 < x \leq 0 \\ (0.125,0,0.25) & 0 < x < 0.5 \end{cases}$$

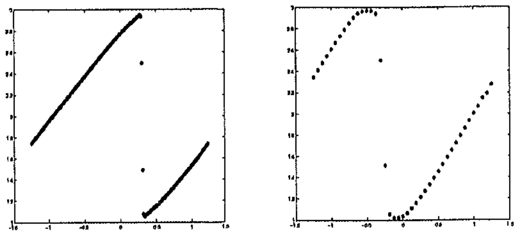

Fig. 20: *=PHM,o=PPHM,left n=256,m=400,right n=40,m=40,CFL=0.2

In this experiments we see as the WENO schemes produce artificial oscillations when the parameter h is not small enough.

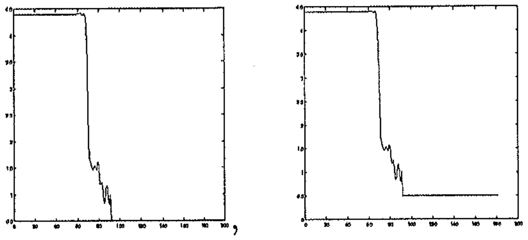

Fig. 21: n=180,m=30,CFL=0.4,left PPHM5,right WENO5

In figures 22-30 we consider Burgues equation with initial data $((-5,5))$:

$$u(x,0)=\begin{cases} 0 & |x|\leq 0.3 \\ \sin(\pi\cdot\frac{(x-0.3)}{0.6}) & else \end{cases}$$

In figure 22 we can see the good resolution of WENO scheme. In figures 23-30 we consider differents CFL numbers to test the limitations of the schemes. The third order modified polynomic scheme is the less sensitive to the CFL condition. We are investigating a fifth order modified scheme.

In figures 33-34 we analyze Euler equations in 1-D, with periodic initial data $2+\sin(4\cdot\pi\ x)$ in $(-0.5,0.5)$.

$$w_t+f(w)_x=0$$

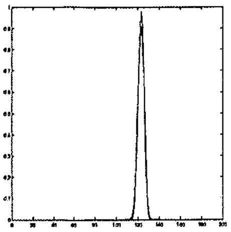

Fig. 22: n=180,m=40,CFL=0.8, WENO5,ENO3,PPHM*,PHM

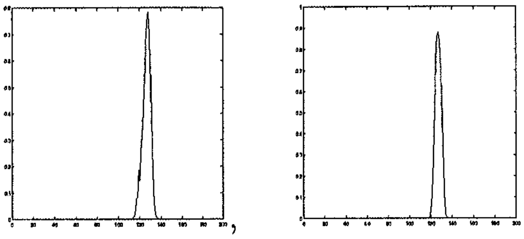

Fig. 23: n=180,m=40,CFL=0.9,left ENO3,right PPHM*

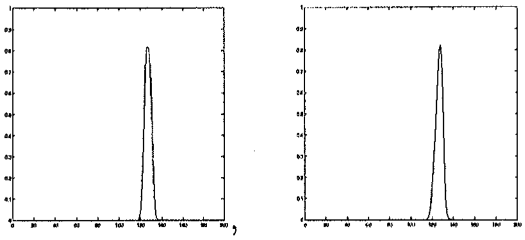

Fig. 24: n=180,m=40,CFL=0.9,left PHM,right WENO5

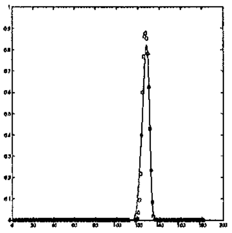

Fig. 25: n=180,m=40,CFL=0.9,o=PPHM*,-=WENO5

where $w = (\rho, m, E)^t$, $\rho = \mathit{density}$, $m = \mathit{momentum}$ and $E = \mathit{energy}$; $f(w) = u \cdot w + (0, P, Pu)^t$, where $u = m/\rho = \mathit{velocity}$ and $P = \mathit{pressure}$. Moreover, it is

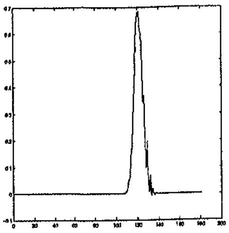

Fig. 26: n=180,m=40,CFL=1.2,PHM

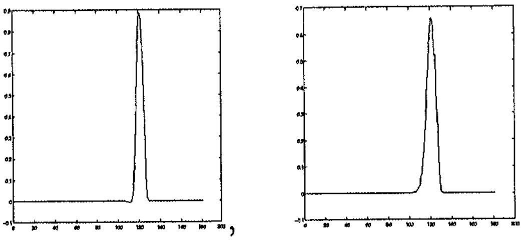

Fig. 27: n=180,m=40,CFL=1.2,left PPHM*,right WENO5

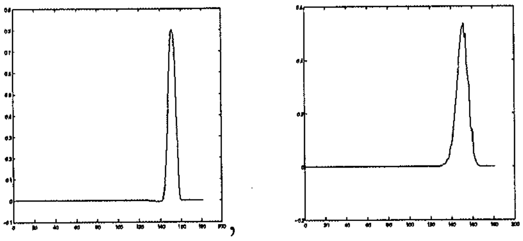

Fig. 28: n=180,m=50,CFL=1.2,left PPHM*,right WENO5

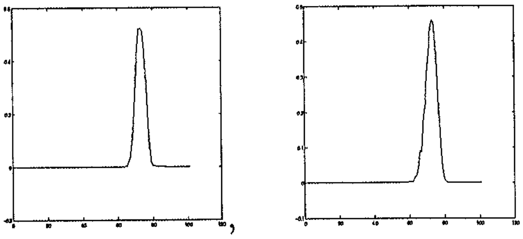

Fig. 29: n=100,m=20,CFL=1.2,left PPHM*,right WENO5

considered the relation

$$P = (\gamma - 1)(E - \frac{1}{2}\rho u^2) \tag{117}$$

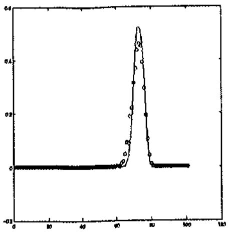

Fig. 30: n=180,m=40,CFL=1.2,o=WENO5,-=PPHM*

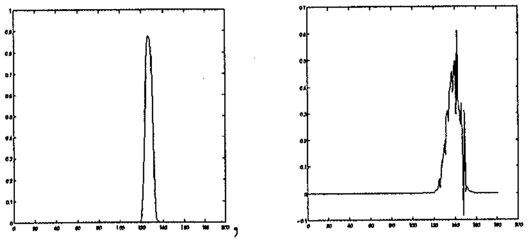

Fig. 31: n=180,m=40,PPHM5,left CFL=0.9,right CFL=1.2

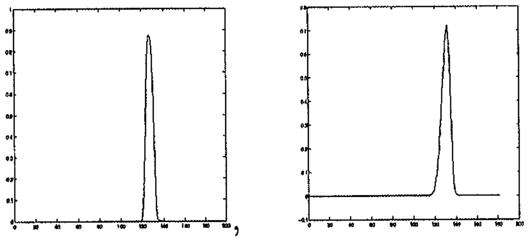

Fig. 32: n=180,m=40,CFL=1,left PPHM5,right WENO5

in our experiments $\gamma = 1.4$.

Notice that in the case of smooth regions all the three algorithms (PHM, PPHM and third order ENO method) are identical, but near of local extremes with the ENO schemes we obtain a better resolution. This pathology of PHM and PPHM can be solve using Yang artificial compression method in the version of Shu and Osher (see [56]-[48]).

In figures 35-38, with the same system of equations, we consider the initial

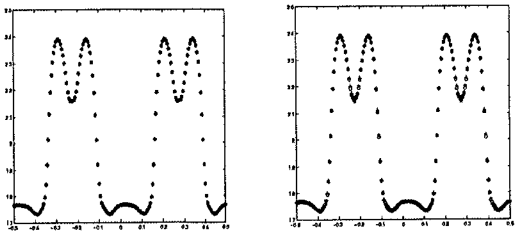

Fig. 33: o=PHM, left +=PPHM, right +=ENO-3, n=180, m=180, CFL=0.4

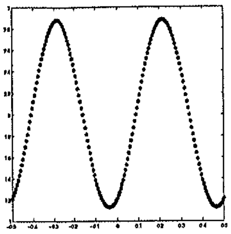

Fig. 34: o=PHM, left +=PPHM, right +=ENO-3, n=180, m=40, CFL=0.4

data:

$$u(x,0) = \begin{cases} (1,0,2.5) & -0.5 < x \leq 0 \\ (0.125,0,0.25) & 0 < x < 0.5 \end{cases}$$

We observe apparent improvements of all algorithms, but with a worse resolution in the ENO scheme, above all when h is large enough.

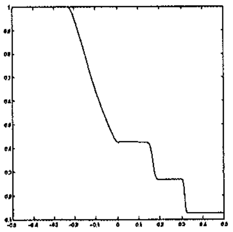

Fig. 35: PHM, n=180, m=80, CFL=0.4

In figures 39-43 we study the initial condition (Euler's equations):

$$u(x,0) = \begin{cases} (4.4,2.3,5) & -0.5 < x \leq -0.3 \\ (p(x),p(x),p(x)) & -0.3 < x < 0 \\ (0.5,0,2.5) & 0 \leq x < 0.5 \end{cases}$$

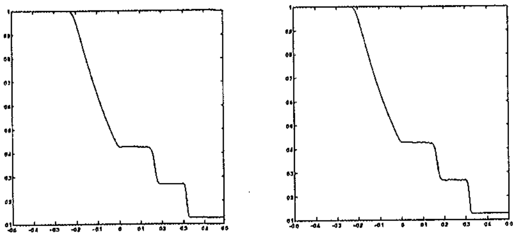

Fig. 36: left PPHM, right ENO-3, n=180, m=80, CFL=0.4

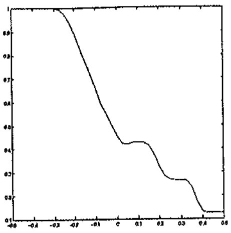

Fig. 37: PHM, n=40, m=20, CFL=0.4

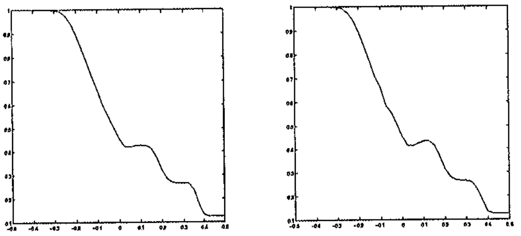

Fig. 38: left PPHM, right ENO-3, n=40, m=20, CFL=0.4

where $p(x) = \sin(40 \cdot \pi \cdot x)$.

Our method is neither sensitive to the CFL number nor to the size of h, on the other hand the ENO schemes are very sensitive.

In figures 44-49 we consider the initial data (Euler's equations):

$$(\rho, v, p) = \begin{cases} (3.857143, 2.629369, 10.33333) & x < -8 \\ (p, 0, 1) & x \geq -8 \end{cases}$$

where $p = 1 + 0.2 \cdot \sin(3 \cdot x)$ in $(-10, 10)$. We observe that the fine structure in the density profile makes the higher order schemes perform much better than the

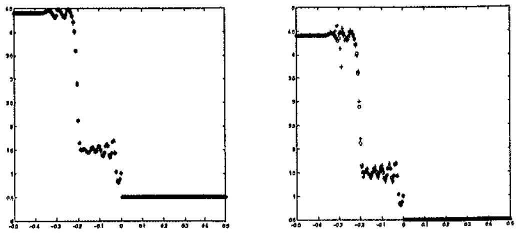

Fig. 39: n=180, m=80, CFL=0.4, o=PHM, left +=PPHM, right +=ENO-3

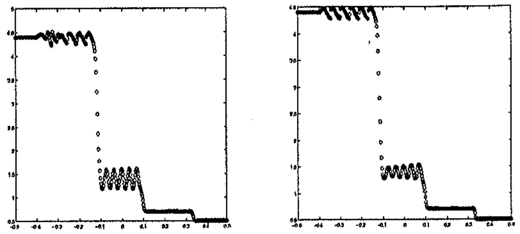

Fig. 40: n=360, m=320, CFL=0.4, left ENO-3, right PHM

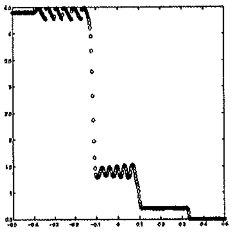

Fig. 41: n=360, m=320, CFL=0.4, PPHM

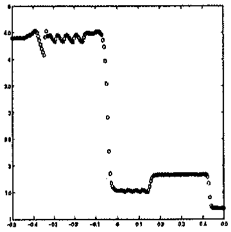

Fig. 42: n=180, m=225, CFL=0.4, ENO-3

lower order methods.

In figures 50-51 we regard linear equation in 2-D in a grid 50 × 50, with initial

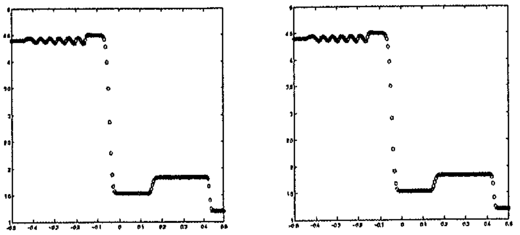

Fig. 43: n=180, m=225, CFL=0.4, left PHM, right PPHM

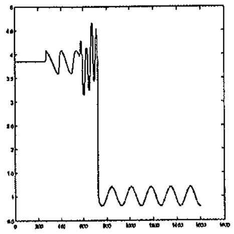

Fig. 44: fine grid,n=1600,m=800,CFL=0.2

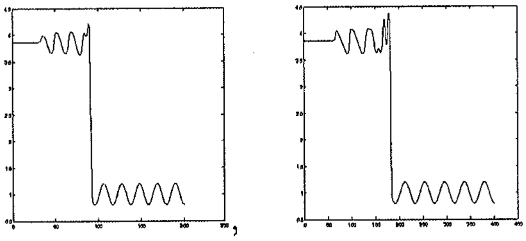

Fig. 45: PHM, left n=200,m=100,CFL=0.2, right n=400,m=200,CFL=0.2

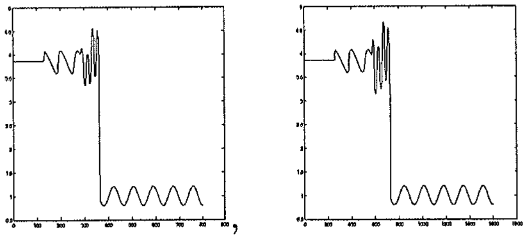

Fig. 46: left PHM,n=800,m=400, right PPHM5,n=200,m=100,CFL=0.2

data $(1, 0, 0.5, 0.8)$.

We have obtained the same conclusion than 1-D test, PHM and PPHM schemes

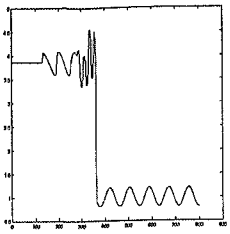

Fig. 47: PPHM5,n=400,m=200,CFL=0.2

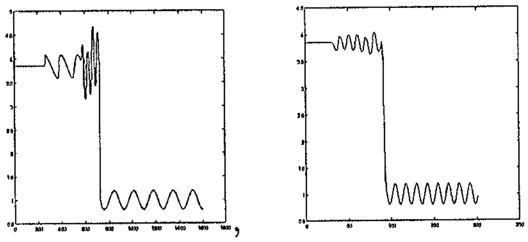

Fig. 48: left PPHM5,n=800,m=400, right WENO5,n=200,m=100,CFL=0.2

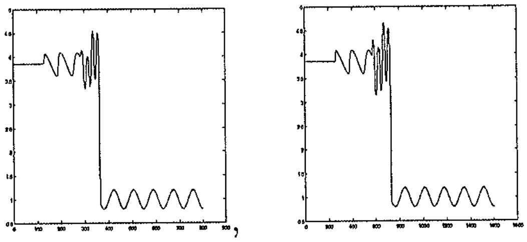

Fig. 49: left WENO5,n=400,m=200, right WENO5,n=800,m=400,CFL=0.2

are very similar.

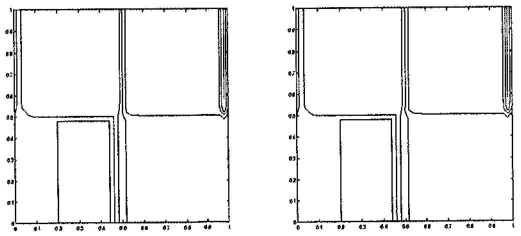

Fig. 50: m=10, CFL=0.2, left PPHM, right PHM

In figures 52-53 we study the linear equation in 2-D with the initial data

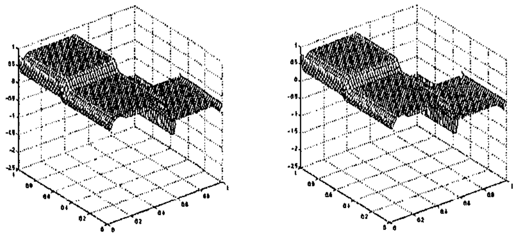

Fig. 51: m=10, CFL=0.2, left PPHM, right PHM

$$u(x,y,0)=\begin{cases} 1 & x^2+y^2<0.25 \\ 0 & else \end{cases}$$

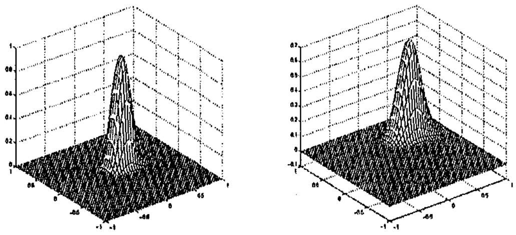

Fig. 52: left m=10, CFL=0.8, right m=30, CFL=0.8, PPHM

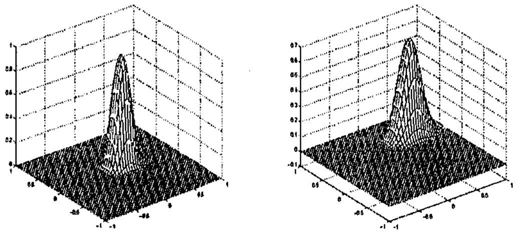

Fig. 53: left m=10, CFL=0.8, right m=30, CFL=0.8, PHM

In the figures 54-55 we analyze Euler equations in 2-D, with same initial data than the shock tube in y-direction and x-direction respectively, we use a grid 60×40 and $CFL = 0.1$.

We observe convergence of PPHM method to the correct entropy solution, with good resolution, for all cases.

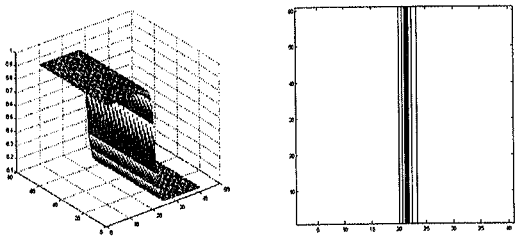

Fig. 54: Shock y-direction, density, PPHM, m=20

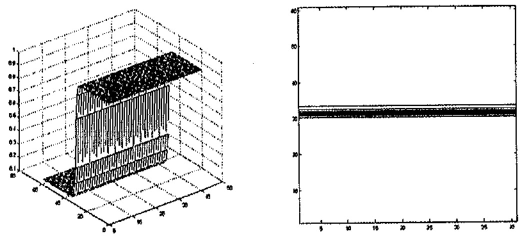

Fig. 55: Shock x-direction, density, PPHM, m=20

In figures 56-59 we consider the initial condition:

$$u(x,y,0) = \begin{cases} (2,0,0,1) & x+y<-1 \\ (a,b,c,d) & x<1 \ |y|<0.75 \\ (0.05,0,0,0.05) & else \end{cases}$$

where $a = 1$, $b = -\sin(\frac{\pi}{6})\ cos(2\cdot\pi\ x\ \cos(\frac{\pi}{6}) + 2\cdot\pi\ y\ \sin(\frac{\pi}{6}))$,

$c = \cos(\frac{\pi}{6})\ cos(2\cdot\pi\ x\ \cos(\frac{\pi}{6})+2\cdot\pi\ y\ \sin(\frac{\pi}{6}))$, $d = \frac{1}{0.4}+0.5\cdot u^2$, $u = \sqrt{(b^2+c^2)}$. We observe that the fine structure in the profiles makes the fifth order scheme perform much better than the third order scheme.

Finally, in the figures 60-61 we consider Euler equations in 1-D, with the initial data:

$$u(x,0) = \begin{cases} (3.86,-0.81,10.33) & 0. < x \leq 0.1 \\ (1,-3.44,1) & 0.1 < x < 1 \end{cases}$$

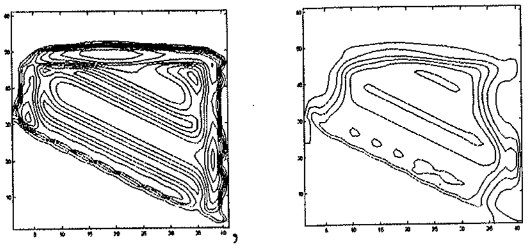

Fig. 56: left Velocity, right Energy,m=40,PPHM

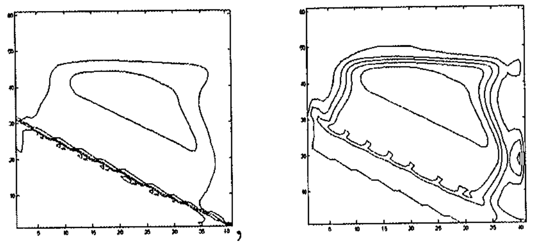

Fig. 57: left Density,right Pressure,m=40,PPHM

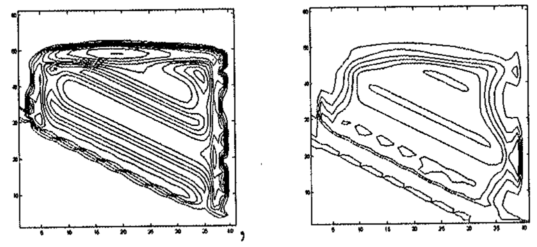

Fig. 58: left Velocity, right Energy,PPHM5,m=40

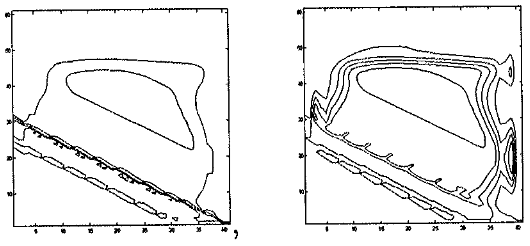

Fig. 59: left Density,right Pressure,PPHM5,m=40

The oscillations of the numeric solution are related with the support of the scheme. If the scheme has a bigger support then it has bigger oscillations.

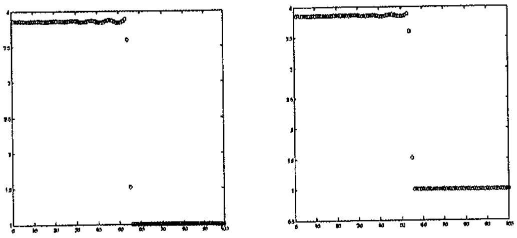

Fig. 60: density,n=100,m=4000,CFL=0.1,left PPHM right PHM

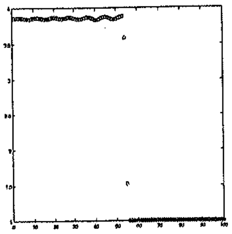

Fig. 61: density,n=100,m=4000,CFL=0.1,ENO-3

Our upwind schemes based on fluxes and the Shu-Osher third order TVD Runge-Kutta method designed for conservation laws seemed to work very well in our preliminary numerical results. As PHM schemes, our methods have two main advantages with respect to the same order ENO (WENO) schemes: they are not very sensitive neither to the CFL number nor to the parameter h and they are localer than ENO-3 in the sense that numerical flux depend on less variables (four and six respectively in order three). Moreover, in presence of discontinuities it is stable and with lower viscosity. The main advantage respect to PHM scheme is the simplicity of our reconstruction, thus, it is less expensive and easily generalizable.

9 Algoritmos de Multirresolución para la Solución Numérica de Leyes de Conservación Hiperbólicas

Dado un esquema en forma conservativa y una apropiada malla uniforme para la solución numérica de un problema de valores iniciales basado en una ley de conservación unidimensional, Harten [34] describe un algoritmo de multirresolución que aproxima a la solución numérica hasta una tolerancia dada de una forma más eficiente. Su representación multi-escala consiste en las medias en celda del nivel más grosero y el conjunto de los errores de predicción.

En una discontinuidad de salto los errores de interpolación permanecen con el mismo tamaño, independientemente del nivel de refinamiento, y esta observación permite identificar la localización de las discontinuidades en la solución numérica. Los flujos numéricos sofisticados y costosos sólo son necesarios en las regiones donde la solución deja o va a dejar de ser suave. La multirresolución permite identificar esas zonas, y así en las zonas no señaladas usar flujos sencillos o interpolación.

La eficacia computacional de estos algoritmos es proporcional al radio de compresión que puede ser analizado para la solución numérica dada por el esquema. Así, en principio, la multirresolución no lineal puede ser una buena herramienta.

La formulación de Harten tiene el inconveniente de trabajar con medias en celdas lo que complica la transferencia entre distintas escalas, sobre todo en dimensiones mayores a uno. Para eliminar estas complicaciones, Shu y Osher [51]-[52] desarrollaron una nueva versión conservativa de los métodos ENO, que usa solamente valores puntuales de las cantidades conservadas.

Los algoritmos siguen los siguientes pasos: se comienza con v^n solución numérica en el paso temporal t_n, calculamos su representación multirresolutiva, evaluamos

$\hat{D}^{n+1}$ ver la nota 1, con esta información decidimos donde interpolamos y donde evaluamos los flujos, finalmente obtenemos v^{n+1}.

Nota 1 *Teniendo en cuenta la compresibilidad y que la propagación tiene velocidad finita Harten propone el siguiente algoritmo que combina el cálculo de $\hat{D}^{n+1}$ con el truncamiento.*

(i) Sea

$$\hat{\imath}(j,k) = 0,\ \ 1 \leq j \leq J_k,\ \ 0 \leq k \leq L. \tag{118}$$

(ii)

```
for $k = L, \ldots, 1$
for $j = 1, \ldots, J_k$
    if $(|d_j^k(v^n)| \leq \epsilon_k)$
```

$$d_j^k(v^n) = 0 \tag{119}$$

else

$$\hat{\imath}(j-l,k) = 1,\ \ -\bar{K} \leq k \leq \bar{K} \tag{120}$$

```
    if $(|d_j^k(v^n)| \leq 2^{p+1}\epsilon_k \ y\ k < L-1)$
```

$$\hat{\imath}(2j-1,k) = 1, \hat{\imath}(2j,k) = 1. \tag{121}$$

(iii) Se define $\hat{D}^{n+1}$ mediante

$$\hat{D}^{n+1} = \{(j,k) : \hat{\imath}(j,k) = 1\}. \tag{122}$$

El rango del parámetro K es

$$1 \leq \bar{K} \leq K \qquad (123)$$

donde K es el soporte del flujo numérico.

También se pueden hacer esquemas híbridos donde la información de la multirresolución es usada para decidir si se usa un esquema no costoso (tipo central) o uno sofisticado.

La idea es usar la compresión de datos de la solución numérica con el objetivo de reducir el coste producido por las evaluaciones del flujo numérico.

Este tipo de algoritmos de multirresolución pueden verse como una alternativa a los métodos malla adaptativos, en los cuales la malla se va adaptando según la solución va evolucionando. La mayor ventaja de la multirresolución respecto a la adaptación es su simplicidad, siendo fácilmente programable e insertable a códigos ya confeccionados.

En esta sección trabajaremos con algoritmos multirresolutivos no lineales. Haremos un pequeño estudio de los mismos comparándolos con los lineales. En principio, con las diferentes multirresoluciones no lineales es posible adaptarse mejor a la estructura de las soluciones, ya que, éstas presentan discontinuidades aisladas, para las que los esquemas no lineales dan mejores resultados. En nuestros experimentos, vemos que esto no siempre se dará. El problema es la difusión introducida por los métodos numéricos. Además, como ya vimos, en 2-D la estructura de la solución puede ser demasiado compleja. Hemos trabajado con multirresolución por promedios en celda.

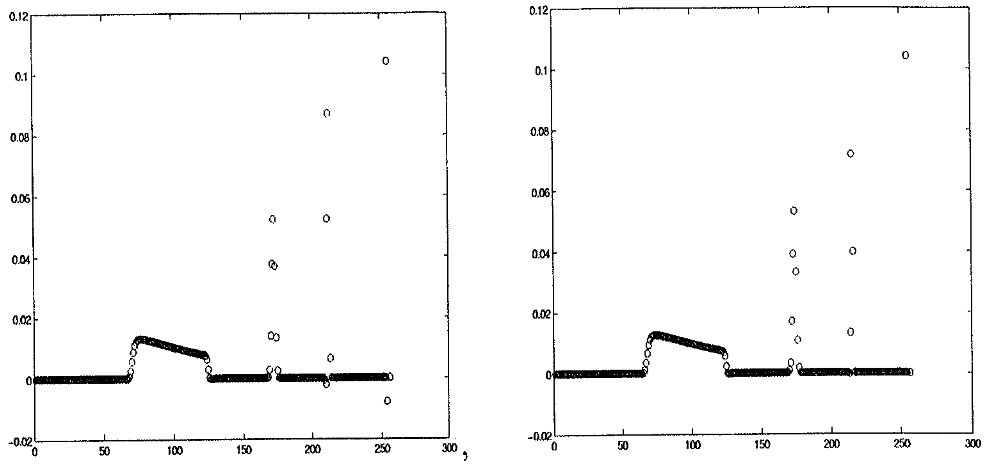

Fig. 62: Detalles, n=512, m=240, CFL=0.4, $tol = 5 \cdot 10^{-4}$, izqda LINEAL, dcha ENO

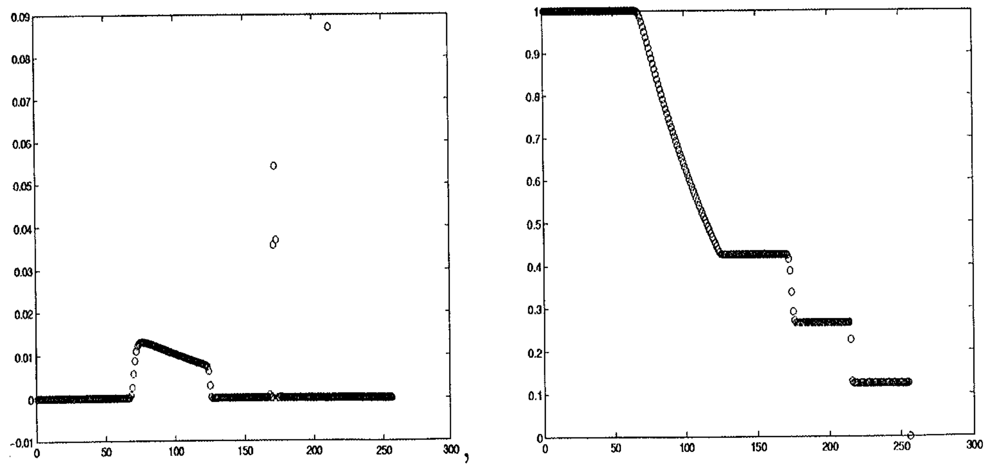

Fig. 63: Detalles,n=512, m=240, CFL=0.4, $tol = 5 \cdot 10^{-4}$, izqda Subcell-Re., dcha Solución Numérica

10 Our present work: A nonlinear scheme for image compression

We already know that an application of multiresolution decompositions is their compression capabilities. A multiresolution representation of an image can be compressed with some loss-control of information.

In the case study [5], we examine the compression properties of ENO schemes. When we consider geometric images, we observe that the compression rate attained by the ENO algorithm is superior to that of the corresponding BOW (Biorthogonal wavelets) scheme. In the case of the ENO-SR scheme, the reduction is even more impressing. On the other hand, the behavior of the three schemes for real (noise) image is absolutely comparable (we refer to [5] for more details), but the nonlinear (ENO, ENO-SR) case is a slight loss in efficiency.

In this section a nonlinear multiresolution algorithm in 2-D is presented. The main idea of this algorithm is a stencil selection procedure that attempts to choose the stencil within a region of "smoothness" of the image. We will introduce a nonlinear interpolation that adapts better to the complex geometries of the real images.

The idea is the following: Given a picture (set of pixels), we apply an edges detector. Once we have these edges, we apply a multiresolution algorithm keeping a representation that it uses less information. Next, we will apply an error control multiresolution algorithm in two dimensions. In this algorithm we will use the information of the detector algorithm to obtain the new nonlinear reconstruction.

10.1 Description of the algorithm

We start with a matrix A (photo). We apply some edges detector algorithm to the matrix. Assume we have in B these edges, we obtain a multiresolution representation (MB) of B, considering B as a set of curves, i.e., using 1-D multiresolution in each direction of the matrix parametrization.

$$B : [0,1] \to R^3 \quad t \to (x(t), y(t), A(x(t), y(t))) \tag{124}$$

In fact, we only need the edges points. It is not necessary to know the complete parametrization.

Then we have the set of the edges points $(x_i, y_i, A(x_i, y_i))$ such that $(x_i, y_i) \in B$, then we ordered the points, finally we apply 1-d multiresolution in each component and we obtain MB. We truncate MB and after the decoding algorithm one obtains $\hat{B}$ $(\hat{B} \simeq B)$.

Remark 10.1 *There are several algorithmn in order to obtain an arrangement of the edges points* $(A_{n\times n})$.

Remark 10.2 *We can use whichever edge detector algorithm.*

Next, we are going to consider a 2-D (non tensor-product and nonlinear) multiresolution. We use a reconstruction that uses information from regions between edges only. Stencils aren't crossing the edges, if it's possible. In order to assure this we have to consider a reconstruction with less order sometimes. We will use the interpolation of the following elementary lemma.

Lemma 10.1 *Let* $y_0, y_1, \ldots, y_m$*;* $(y_i = y_j$ *if and only if* $i = j)$ *and for each* $k = 0, 1, \ldots, m$ *the values* $x_0^k, x_1^k, \ldots, x_m^k$*;* $(x_i^k = x_j^k$ *if and only if* $i = j)$ *and support* *ordinates $f_{i,k}$; $i = 0, 1, \ldots, n_k$; $k = 0, \ldots, m$. Suppose without loose of generality that the y_k are numerate in such a fashion that*

$$n_0 \geq n_1 \geq n_m.$$

It can prove by induction over m that exactly one polynomial:

$$p(x, y) = \sum_{\mu=0}^{m} \sum_{\nu=0}^{n_\mu} \alpha_{\nu,\mu} x^\nu y^\mu \tag{125}$$

exists with

$$p(x_i^k, y_k) = f_{i,k}$$

$i = 0, 1, \ldots, n_k$; $k = 0, 1, \ldots, m$

In order to avoid problems with the stability we are going to use an error control encoding in the multiresolution algorithms. In this algorithm, we have to specify what is the reconstruction P_{k-1}^k. In our case, P_{k-1}^k is based on the nonlinear reconstruction defined as follows: If we have to approximate some point $(x, y) \in A$, we distinguish two cases:

I) $(x, y) \in \hat{B}$ then "anything to do", because we are going to keep $\hat{B}$.

II) $(x, y) \in A \backslash \hat{B}$ then we build a polynomial of degree μ ($\mu \leq 4$) using an stencil within a region of smoothness of the image (if it is possible).

At the end, we only need the following information:

$$\{\bar{f}_0, \hat{e}_{i,j}^k \ s.t. \ (x_i^k, y_j^k) \in A \backslash Edges, \hat{B}\}$$

10.2 Compression of edges

We consider the edges of three figures 64-65. In table 12 and figures 66-68 we analyze our scheme. The results are impressive.

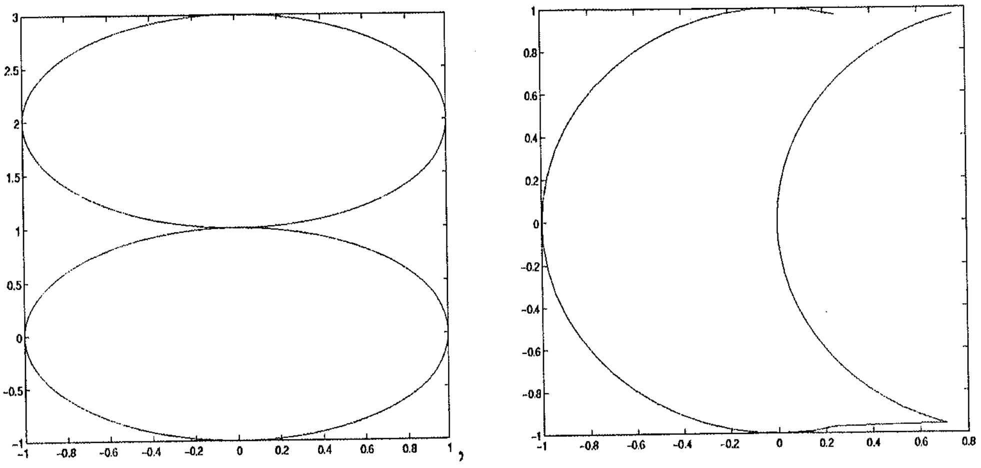

Fig. 64: left edges 1, right edges 2

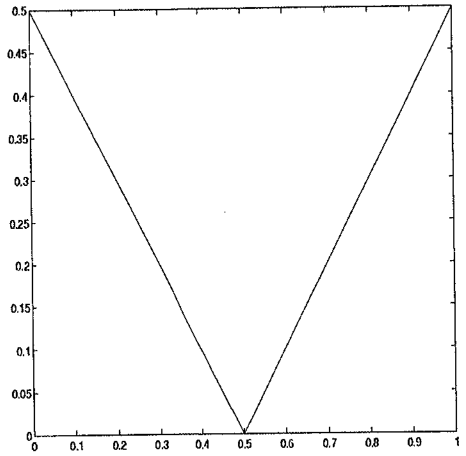

Fig. 65: edges 3

10.3 Some examples in 2-D

We start with the geometric figure 69

$$A(x,y) = \begin{cases} 0 & if\ x \in \Omega_1 \\ 256 & if\ x \in \Omega_2 \end{cases} \tag{126}$$

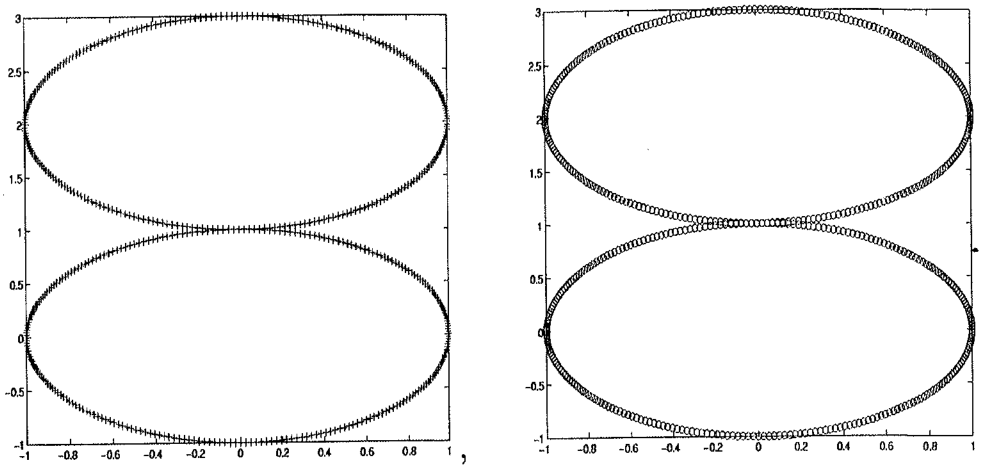

Fig. 66: left o=aproximation edges 1,right +=exact edges 1,n=513

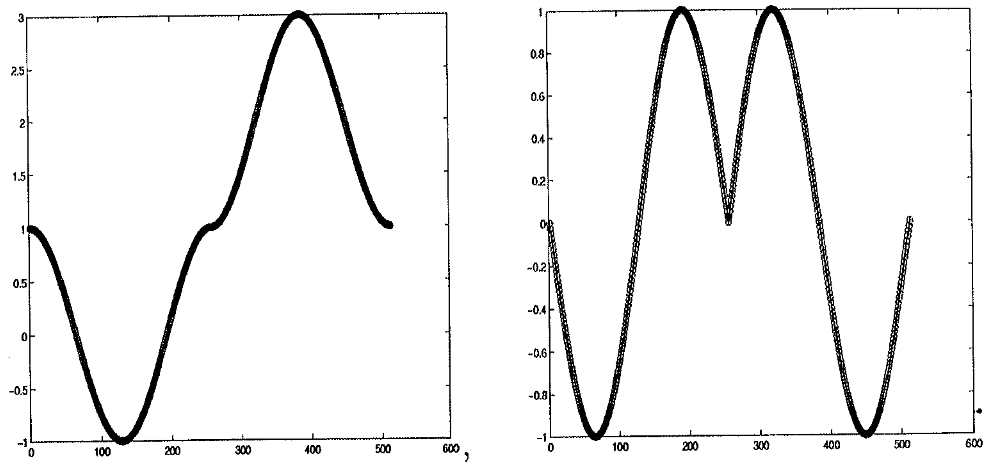

Fig. 67: o=aprox., +=exact edges 1, left x-, right y-direction,n=513

where $\Omega_1 = [0.25, 0.75] \times [0.25, 0.75]$ and $\Omega_2 = [0, 1] \times [0, 1] \setminus \Omega_1$. We use a reconstruction of degree 4 in 1-D and of degree 2 in 2-D. In table 13 we can see the good properties of our scheme. It obtains better results than linear and ENO-SR schemes see [5].

Finally, we will analyze our scheme in a real image. We will consider a Varda's vertical cut, see fig. 70. If we use information from "smooth" regions, the compre-

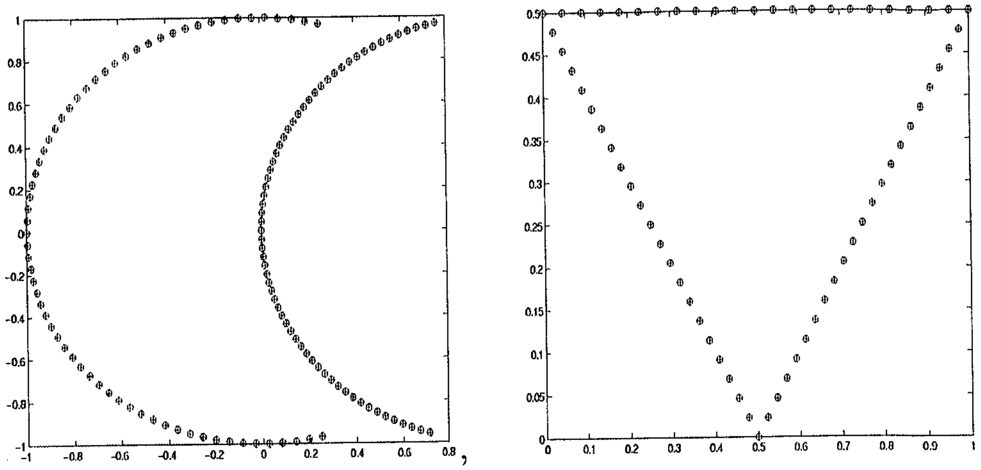

Fig. 68: o=aproximation, left +=exac edges 2 and n=129, right +=exact edges 3, n=65

	edges 1	edges 2	edges 3
L_∞-error	0.	$1.7 \cdot 10^{-3}$	$3 \cdot 10^{-4}$
Ratio-Compression	7.22	5.16	7.77
Number of non zeros	0	8	1
Size of grid	65	129	513

Table 12: tol=0.002,4 scales

ssion will be bigger. We use the simplest pointvalues reconstruction with 4 points. In table 14 we display the error. The adapted method gives better results near the singularities.

At present we are working in the full implementation of this non-linear scheme.

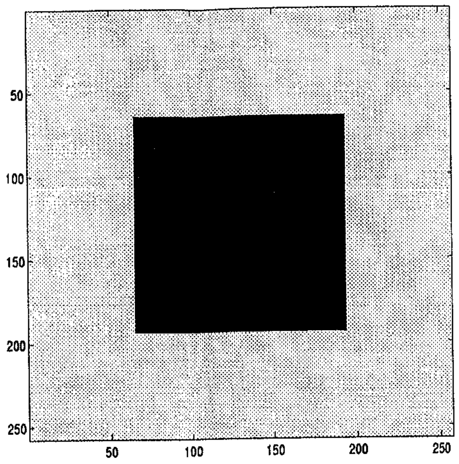

Fig. 69: Geometric figure, 257 × 257

Scales	Details non zeros	Error
2	8	0
3	12	0
4	16	0

Table 13: 257 × 257 points, nonlinear scheme

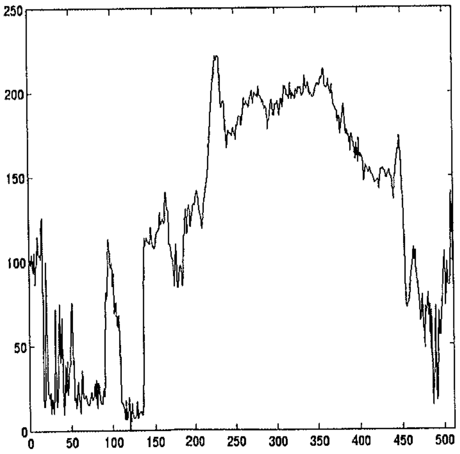

Fig. 70: cut, column=284

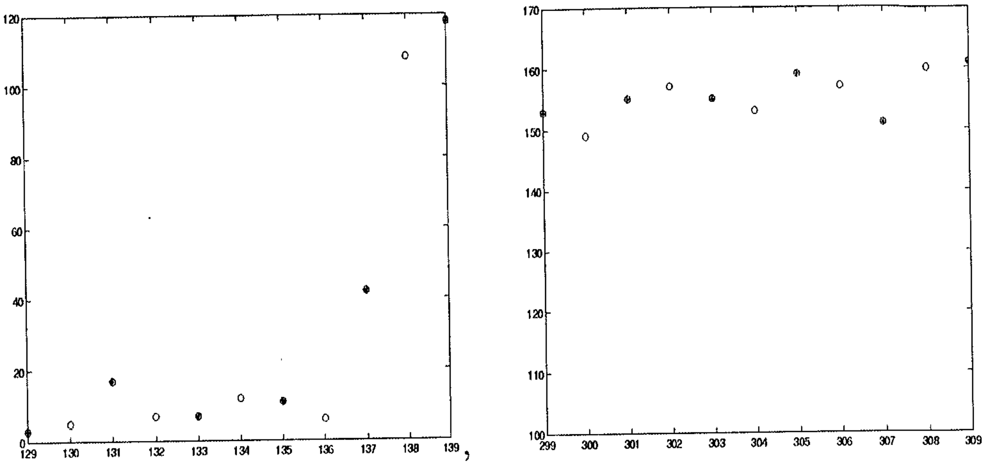

Fig. 71: left zoom near i=136, right zoom near i=304

	Adapted	Centered
near singularity, i=136	3.5	16
“smmoth regions ”, i=304	4.12	4.12

Table 14: Error,(in real image *noise* ≈ 5)

11 Conclusiones, perspectivas y algunas elucubraciones

A lo largo de este trabajo se han estudiado algunos aspectos de la Multirresolución á la Harten. Se han considerado diversas aplicaciones, profundizando en la parte numérica de las mismas. No obstante, quedan cuestiones interesantes que intentaremos ir completando en sucesivos trabajos.

En la cuarta sección hemos introducido un detector de discontinuidades, basado en diferencias divididas. Hemos adaptado nuestro algoritmo a la presencia de ruido. Los resultados numéricos fueron satisfactorios, encontrándose las discontinuidades reales de la señal y despreciando las ficticias producidas por el ruido. Además las adaptaciones introducidas, pueden servir para obtener una multirresolución no lineal aplicable a señales con ruido, como pueden ser las imágenes reales.

En la siguiente sección estudiamos el concepto de estabilidad para el caso no lineal. Hemos introducido algoritmos de error control en 2-D a partir de los cuales podemos obtener cotas explícitas (a priori y a posteriori) del error. Aunque dichos algoritmos nacen para poder estabilizar las multirresoluciones no lineales, son aplicables al caso lineal, para los que se tendría también cotas explícitas del error sin constantes no "calculables" como aparecen en los teoremas clásicos de la teoría de Wavelets. Los algoritmos introducidos sirven para cualquier tipo de multirresolución, recuperándose como caso particular el producto tensorial que desarrollamos en [4]. Hemos introducido diferentes normas y algoritmos estables respecto a las mismas.

En principio, en los algoritmos de error-control no se tiene información del radio de compresión. No obstante, cabe destacar, que en la práctica, el error-control si está confirmado como una buena herramienta de compresión (ver [4]). Por otro lado, es posible obtener, en algunos casos, cotas del número de no ceros de estos algoritmos y por lo tanto tener control de la compresión.

Caso Interpolatorio

Denotamos el error acumulado de compresión al nivel k mediante ${\varepsilon_j}^k$.

$$ {\varepsilon_j}^k := \bar{f}_j^k - \hat{f}_j^k $$

En [33], se prueba que $|{\varepsilon_j}^k| < \epsilon_{k-1}$. $(\epsilon_{k-1} \geq \epsilon_{k-2} \geq \epsilon_{k-3} \geq \ldots)$.

En este caso, tenemos:

$$ \bar{f}_{j+\frac{1}{2}}^k - \frac{9}{16}(\hat{f}_j^{k-1} + \hat{f}_{j+1}^{k-1}) + \frac{1}{16}(\hat{f}_{j-1}^{k-1} + \hat{f}_{j+2}^{k-1}) = $$

$$ \bar{f}_{j+\frac{1}{2}}^k - \frac{9}{16}(\bar{f}_j^{k-1} + \bar{f}_{j+1}^{k-1}) + \frac{1}{16}(\bar{f}_{j-1}^{k-1} + \bar{f}_{j+2}^{k-1}) + \frac{9}{16}({\varepsilon_j}^{k-1} + {\varepsilon_{j+1}}^{k-1}) - \frac{1}{16}({\varepsilon_{j-1}}^{k-1} + {\varepsilon_{j+2}}^{k-1}) $$

Así

$$ |\hat{d}_j^k| \leq |{d_j}^k| + \frac{10}{16}(2\epsilon_{k-1}) $$

Si $\hat{d}_j^k > \epsilon_k$, *entonces* $\epsilon_k < |d_j^k| + \frac{20}{16}\epsilon_{k-1}$ *por lo tanto*

$\epsilon_k - \frac{20}{16}\epsilon_{k-1} < |d_j^k|$

Tomando $\epsilon_L := \epsilon$ *y* $\epsilon_{k-1} := \frac{16}{20}\frac{\epsilon_k}{c}$, $c > 1$

Si $|\hat{d}_j^k| > \epsilon_k$, *tendremos* $|d_j^k| > (1-\frac{1}{c})\epsilon_k$, *y así cardinal*$\{|\hat{d}_j^k| > \epsilon_k\} \leq$ *cardinal*$\{|d_j^k| > (1 - \frac{1}{c})\epsilon_k\}$

Ahora bien, si f es suave de la teoría de los wavelets, se tiene cardinal$\{|d_j^k| > (1 - \frac{1}{c}\epsilon_k)\} \leq M$

Por lo tanto tendremos control del número de coeficientes significativos y así de la compresión.

Caso de medias en celda

En este caso, con las mismas ideas, se puede obtener:

$$|\hat{d}_j^k| = |d_j{}^k| + \varepsilon_j^{k-1} - \frac{22}{128}(\varepsilon_{j+1}^{k-1} - \varepsilon_{j-1}^{k-1}) - \frac{3}{128}(\varepsilon_{j+2}^{k-1} - \varepsilon_{j-2}^{k-1})$$

Así

$$|\hat{d}_j^k| \leq |d_j{}^k| + \epsilon_k + \frac{25}{128}(2\epsilon_k) =$$
$$|d_j{}^k| + \frac{178}{128}\epsilon_k$$

Y obtemos también control.

Para finalizar los comentarios sobre la estabilidad no lineal, daremos una posible alternativa al error control. Nos centraremos en el caso extremo de "subcell resolution". La idea es aplicar un detector de singularidades (adaptado al ruido, como el introducido en la tercera sección). Una vez determinadas las discontinuidades (con una cierta precisión) en el nivel más grosero, extrapolar los polinomios interpoladores hasta dicha singularidad (i.e. aplicar "subcell resolution"), mantener esta extrapolación hasta llegar a un nivel con una discretización h_k tal que el lugar de la discontinuidad encontrada no sea preciso. Llegado a este momento se aplicaría de nuevo el detector para precisar el lugar de la discontinuidad y se volvería a aplicar los pasos anteriores. El algoritmo es quasi-lineal.

En la novena sección estamos estudiando un algoritmo no lineal para la compresión de datos. La idea es usar interpolación adaptada la cual sólo utilice información de regiones de "suavidad" de la función. Como región de suavidad, entendemos las que no contienen ejes de la imagen. Estos puntos una vez obtenidos se almacenan de forma comprimida, lo cual hace aumentar la compresión final en gran medida, haciendo altamente competitivo nuestro método. Además la adaptación resulta correcta, ya que, esta sólo se produce cerca de los ejes y no como con los esquemas tipo ENO en los que la adaptación se produce por todas partes al detectar las discontinuidades ficticias producidas por el ruido propio que contienen las imágenes (ver [5]).

Otro esquema no lineal, pero mejor adaptado que el ENO, sería hacer interpolación central en todos los sitios salvo que el detector adaptado encontrara una singularidad. Este esquema sería una modificación del ENO, con la diferencia de usar un detector que si está adaptado al ruido y así sólo variar la central en los lugares correctos.

En la sección seis extendemos al caso no lineal los esquemas de "wavelets-packets". Hemos introducido algoritmos de error control y hemos comprobado las mejoras que, en ciertos casos, proporcionan dichos esquemas. La propiedad fundamental de la "multiresolution-packets-nolineal" es una adaptación total, i.e., proporciona representaciones que se adaptan bien a "tiempo-frecuencia" (como los "wavelets-packets") y a las discontinuidades (como la multirresolución no lineal).

En la sección octava estudiamos leyes de conservación. Hemos desarrollado esquemas de alto orden aplicando las ideas introducidas por Marquina en [48]. Se han estudiado teórica y numéricamente, comparando su eficacia con los métodos de alto orden PHM, ENO y WENO. Los resultados fueron altamente satisfactorios.

Por último hemos relacionado las leyes de conservación con la multirresolución. Hemos presentado los esquemas lineales de Harten, hemos introducido esquemas no lineales y los hemos comparado. No está clara la mejora de los esquemas no lineales (que son más costosos), ya que, a pesar de la aparición expontánea de discontinuidades en la solución de las leyes de conservación, los métodos numéricos llevan asociada difusión y así las discontinuidades no aparecen "limpias" reduciéndose la eficacia de la adaptación no lineal.

Como futuras lineas de trabajo destacamos:

-Implementación de un buen compresor para imagenes reales.

-Aplicación de los algoritmos WP desarrollados, a distintas áreas de conocimiento: compresión de imágenes, procesamiento de señales, aceleración de métodos para ecuaciones...

-Aplicación de los métodos PPHM a las ecuaciones de Hamilton-Jacobi.

-Aplicación a sistemas de leyes de conservación en 2-D del método PPHM-5 (se realizará en una estancia postdoctoral en Marsella).

References

[1] R. Abgrall, *Design of an Essentially Nonoscillatory Reconstruction Procedure on Finite Element Type Meshes*, February 1992, ICASE Report 91-84, December 1991, revised INRIA report 1592.

[2] I. Alonso-Mallo, *Single step methods with optimal order of convergence for partial differential equations.* To appear in Appl. Num. Math.

[3] S. Amat, *Un estudio comparativo sobre la estabilidad en la extrapolación de diferentes métodos de resolución de EDO's.* Tesis de Licenciatura. Dept.Mat.Apl.Univ.Valencia, 1998.

[4] S.Amat, F.Aràndiga, A.Cohen and R.Donat. *Tensor product multiresolution analysis with error control for compact image representation.* To appear in Signal processing 2001.

[5] S.Amat, F.Aràndiga, A.Cohen, R.Donat, G.García and M.von Oehsen *Data compresion with ENO schemes: A Case Study.* To appear in ACHA 2001.

[6] F. Aràndiga and V. Candela, *Multiresolution Standard Form of a Matrix*, SIAM J. Numer. Anal., 33, pp. 417-434, 1996.

[7] F. Aràndiga, V. Candela and R. Donat, *Fast Multiresolution Algorithms for Solving Linear Equations: A Comparative Study*, SIAM J. Sci. Comput., 16, pp. 581-600, 1995.

[8] F.Aràndiga and R.Donat, *Nonlinear Multi-scale Decompositions: The Approach of A.Harten.* Numerical Algorithms. V 23 175-216, 2000.

[9] F.Aràndiga, R. Donat and A. Harten, *Multiresolution Based on Weighted Averages of the Hat Function I: Linear Reconstruction Operators*, SIAM J. Numer. Anal., 36 (1), pp. 160-203, 1999.

[10] F.Aràndiga, R. Donat and A. Harten, *Multiresolution Based on Weighted Averages of the Hat Function II: Nonlinear Reconstruction Operators*, SIAM J. Sci. Comput., 20 (3), pp. 1053-1093, 1999.

[11] E. Bacry, S. Mallat and G. Papanicolau, *A Wavelet Based Space-Time Adaptive Numerical Method for Partial Differential Equations*, Math. Modelling and Numer. Anal., 26, pp. 703-834, 1992.

[12] R.H. Bamberger *A Method for Image Interpolation Based on a Novel Multirate Filter Bank Structure and Properties of the Human Visual System*, Proc. of SPIE, vol. 1657 (Image Processing Algorithms and Techniques), pp.351-362, 1992.

[13] G. Beylkin, *Wavelets, Multiresolution Analysis and Fast Numerical Algorithms*, INRIA lectures, manuscript, 1991.

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

[15] A. Cavaretta, W. Dahmen and C. Micchelli *Stationary Subdivision*, AMS Memoirs, 453, 1991.

[16] A.K. Chan, C.K. Chui, J.Zha and Q. Liu *Local cardinal spline interpolation and its application to image proccessing*, Proc. of SPIE, vol. 1610 (Curves and Surfaces in Computer Vision and Graphics), pp.272-283, 1991.

[17] G. Chen and Rui J.P. de Figueiredo *A Unified Approach to Optimal Image Interpolation Problems Based on Linear Partial Differential Equations*, IEEE Trans. on Image Processing. vol. 2, no. 1, pp. 41-49, 1993.

[18] A. Cohen, I. Daubechies and J.C. Feauveau, *Biorthogonal Bases of Compactly Supported Wavelets*, Comm. Pure Applied Math., 45, pp. 485-560, 1992.

[19] R.R.Coifman, Y.Meyer, Y.Quake, M.V.Wickerhauser, *Signal processing and compression with wavelet packets*, 'Progress in Wavelet Analysis and Applications ', Y.Meyer and S.Roquesed., Editions Frontières, 1993.

[20] R.R.Coifman, Y.Meyer, M.V.Wickerhauser: *Size properties of the wavelet packets*, 'Wavelets and their applications', Ruskai et al. (ed.), Jones and Barlet, pp. 453-470, 1992.

[21] R.R.Coifman, M.V.Wickerhauser, *Entropy-based methods for best basis selection*, IEEE Trans. on Inf. Theory, 28, 2, 719-746, 1992.

[22] P.Colella and P.R.Woodward, *The piecewise parabolic method for gas dinamics*, J.Comput. Phys., 54, pp.174, 1984.

[23] G. Dahlquist & A. Björk , *Numerical Methods.* Prentice-Hall, inc. 1974.

[24] I. Daubechies *Ten Lectures on Wavelets*, CBMS-NSF Regional Conference Series in Applied Mathematics 1992.

[25] I. Daubechies *Orthonormal bases of compactly supported wavelets*, Comm. Pure and Appl. Math., 41, pp. 909-996, 1988.

[26] G. Deslauriers and S. Duboc, *Symmetric Iterative Interpolation Scheme*, Constr. Approx., 5, pp. 49-68, 1989.

[27] R. Donat *Studies on Error Propagation for Certain Nonlinear Approximations to Hyperbolic Equations: Discontinuities in Derivatives*, SINUM, 31, pp. 655-679, 1994.

[28] R.Donat and A.Marquina , *Capturing Shock Reflections: An Improved Flux Formula*, J.Comput. Phys., 25, pp.42-58, 1996.

[29] N. Dyn, J.A. Gregory and D. Levin *Analysis of Linear Binary Subdivision Schemes for Curve Design*, Constr. Approx., 7, pp. 127-147, 1991.

[30] E. Hairer & G. Warner , *Solving Ordinary Differential Equations II.* Board, 1991.

[31] A. Harten, *Discrete Multiresolution Analysis and Generalized Wavelets*, J. Appl. Numer. Math., 12, pp. 153-192, 1993.

[32] A. Harten, *Multiresolution Representation of Data*, UCLA CAM Report, pp. 93-13, 1993.

[33] A. Harten, *Multiresolution Representation of Data II: General Framework*, SIAM J. Numer. Anal. 33 (3), pp 1205-1256, 1996.

[34] A. Harten, *Multiresolution Algorithms for the Numerical Solution of Hyperbolic Conservation Laws*, Comm. Pure Appl. Math., 48 (12), pp 1305-1342, 1995.

[35] A.Harten, *High resolution schemes for hyperbolic conservation laws*, J.Comput.Phys., 49, pp.357-393, 1983.

[36] A.Harten, *On a class of high resolution total variation stable finite difference schemes*, SIAM J.Numer.Anal., 21, pp.1-23, 1984.

[37] A.Harten and S.J.Osher, *Uniformly high order accurate nonoscillatory schemes I*, SIAM J. Numer.Anal., 24, pp. 279-309, 1987.

[38] A.Harten, S.J.Osher, B.Engquist and C.Chakravarthy, *Some results on uniformly high-order accurate essentially non-oscillatory schemes*, Appl. Numer. Math., 2, pp. 347-377, 1987.

[39] A. Harten, B.Engquist, S. Osher and S. Chakravarthy, *Uniformly High Order Accurate Essentially Non-Oscillatory schemes III.* Journal Comput. Phys., 71, pp 231-303,1987.

[40] A. Harten and I. Yad-Shalom *Fast Multiresolution Algorithms for Matrix-Vector Multiplication*, SIAM J. Numer. Anal., 31, pp. 1191-1218, 1994.

[41] Y.S. Ho and A. Gersho *Variable-rate contour-based interpolative vector quantization for image coding*, Proc. IEEE Intl. Conf. on ASSP, pp.750-754, 1988. SPIE, vol. 1610, pp.272-283, 1991.

[42] E. Hairer & G. Warner , *Solving Ordinary Differential Equations II.* Board, 1991.

[43] P.Joly, Y.Maday, V.Perrier, *Towards a Method for Solving Partial Differential Equations by Using Wavelet Packet Bases*, Comput. Methods in Appl. Mech. and Engrg. 116, pp.301-307, 1994.

[44] D. Kahaner , *Numerical Methods and Software.* Prentice-Hall, 1989.

[45] J.D. Lambert , *Numerical Methods for Ordinary Differential Systems.* Wiley, 1991.

[46] R.J. Leveque. *Numerical Methods for Conservation Laws.* Birkhäuser Verlag (Lectures in Mathematics), 1990.

[47] X-D. Liu, S. Osher and T. Chan. *Weighted essentially non-oscillatory schemes.* J. Comput. Phys., 115, pp.200-212, 1994.

[48] A.Marquina, *Local piecewise hyperbolic reconstruction of numerical fluxes for nonlinear scalar conservation laws*, SIAM J.Sci.Comput., 15 (4), pp.892-915, 1994.

[49] S.J.Osher and C.Chakravarthy, *High resolution schemes and the entropy condition*, SIAM J.Numer.Anal., 21, pp. 955-984, 1984.

[50] P.L.Roe , *Approximate Riemann solvers, parameters vectors, and difference schemes*, J. Comput. Phys., 43, pp. 357-372, 1981.

[51] C.W.Shu and S.J.Osher, *Efficient implementation of essential non-Oscillatory shock capturing schemes*, J.Comput.Phys., 77, pp.231-303, 1987.

[52] C.W.Shu and S.J.Osher, *Efficient implementation of essential non-Oscillatory shock capturing schemes II*, J.Comput.Phys., 83, pp.32-78, 1989.

[53] J. Stoer & R. Burlirsch , *Introduction to Numerical Analysis.* Springer-Verlag, 1993.

[54] J.L. Van Iwaarden , *Ordinary Differential Equations with Numerical Techniques.* Harcourt Brace Jovanovich publishers, 1985.

[55] S. Thurnhofer, M. Lightstone and S.K. Mitar *Local cardinal spline interpolation and its application to image proccessing*, Proc. of SPIE, vol. 2094, pp.614-625, 1993.

[56] H.Yang, *An artificial compression method for ENO schemes. The slope modification method*, J.Comput.Phys., 89, pp. 125-160, 1990.

[57] M.V.Wickerhauser, *Picture compression by best-basis sub-band coding*, prepint, Yale University (New Haven, Connecticut), 1990.