

# SOLVING SHALLOW WATER EQUATIONS BY THE SPARSE POINT REPRESENTATION METHOD

Maibys Sierra Lorenzo\*<sup>1</sup>, Angela León Mecías\*\*<sup>2</sup>, Lourdes Álvarez Escudero\*

\* Centro de Física de la Atmósfera, Instituto de Meteorología, Casa Blanca, La Habana, Cuba

lourdes.alvarez@insmet.cu

\*\* Facultad de Matemática y Computación, Universidad de La Habana, San Lázaro y L CP-10400 La Habana, Cuba,

## ABSTRACT

In this contribution a finite difference scheme (FDS) over a temporally and spatially adaptive sparse mesh is presented. The sparsity of the mesh is achieved using the Sparse Point Representation method, which is based on an interpolating subdivision scheme taking as indicator for the sparse representation the points with wavelet coefficients higher than a given threshold. Our approach is a method for solving time dependent partial differential equations in general, but in this paper, it is tested solving the shallow water equations, which also to the best of our knowledge constitute a new way to solve such equations. For the numerical simulation, a modified leapfrog finite difference scheme is used on the Sparse Point Representation based sparse mesh. The gain in compression and CPU time with respect to the FDS on a uniform mesh for large size meshes is reported. Regarding other adaptive mesh refinement, accuracy improvement is obtained. These facts demonstrate the efficiency of our proposal.

**KEYWORDS:** adaptive solution of partial differential equations, subdivision scheme, interpolating wavelet transform, refinement criteria

**MSC:** 65F50

## RESUMEN

En este trabajo se presenta un esquema en diferencias finitas (EDF) adaptado a una malla que varía en el tiempo y en el espacio, reteniendo pocos puntos. El patrón de los puntos que se conservan, se obtiene mediante el llamado método de Representación Dispersa de Puntos (SPR por sus siglas en inglés), basado en un esquema de subdivisión interpoladora, que toma como indicador para la retención de los puntos, los coeficientes wavelets que exceden un umbral predefinido. Nuestra propuesta es un algoritmo que puede ser usado para resolver ecuaciones diferenciales parciales dependientes del tiempo en general, pero en este trabajo se aplica a la resolución del problema de aguas someras, lo cual según nuestro conocimiento constituye una nueva vía para resolver este modelo. Para la simulación numérica se usa el esquema leapfrog sobre la malla adaptativa según el SPR. Se demuestra la ganancia en compresión y tiempo de ejecución con respecto a un EDF sobre una malla uniforme para grandes dimensiones. Se obtiene una mejoría en precisión con respecto a otros tipos de refinamiento de mallas adaptativo. Estos resultados demuestran la eficiencia de nuestra propuesta.

**PALABRAS CLAVES:** solución adaptativa de ecuaciones diferenciales en derivadas parciales, esquema de subdivisión, transformada wavelet interpoladora, criterio de refinamiento.

**MSC:** 65F50

## 1. INTRODUCTION

Physical processes taking place in the atmosphere occur in different spatial and time scales, from long waves, fronts and hurricanes covering thousands of kilometers and lasting weeks, to severe local storms and turbulences covering some meters for few minutes. Properly handling of spatial scales and time evolution is crucial for the right numerical simulation of PDEs models representing physical processes in the atmosphere, ocean and magnetosphere. The PDEs systems modeling the atmosphere, also known as numerical weather models are classified in two categories: global or general circulation models (GCMs) and regional models (RCMs) <sup>[5, 30, 34]</sup>. RCMs allows us to make forecasts that are more precise in both time and space, for critical human activities. GCMs combined with nested RCMs, have the potential to provide geographically and physically consistent estimates of regional climate change.

---

<sup>1</sup> maibys.lorenzo@insmet.cu,

<sup>2</sup> angela@matcom.uh.cu

Due to limited computational resources, we are not able to model and resolve all phenomena. In the Institute of Meteorology in Cuba, one uses RCMs for weather forecasts assuming uniform dense static meshes to resolve the model equations. This means that, in operational regime, the meshes are uniformly refined (resolution is increased) across the domain but the resolution do not change with time. Nevertheless, a good representation of small-scale processes, require very fine meshes, which increases the computational cost significantly. On the other hand, there are often large portions of the RCMs domains where high levels of refinement are not needed. The use of a fine mesh in such regions represents a waste of computational effort. It is evident that high resolution has to be achieved where it is most required, which is the goal of adaptive mesh refinement.

In this paper, we present a numerical algorithm to solve weather forecasting models with adaptive meshes. The goal is modelling meteorological events taking place in small fractions of the RCMs domains with reduced computational cost. In order to solve numerically the model equations in the RCMs, we propose a modified leapfrog finite difference scheme constructed over a temporally and spatially varying mesh.

In order to simulate small-scale processes with less computational efforts, several approaches called in the literature adaptive strategies (AS) [28], have been developed. Among them we can mention the mesh refinement methods using nested [33, 35] and stretched grids [15], the adaptive mesh refinement (AMR) methods [2, 4, 7, 20, 25] and the multiresolution methods [11, 16-19, 26]. We will pay special attention to methods in the last two categories, which offer a dynamic refinement of the grids.

The AMR scheme allows an improved mesh resolution in regions where and when it is required. Starting from a uniform mesh, one check the local numerical solution errors and modify (refine) the mesh in those regions where this error exceeds a certain threshold. This leads to meshes with high resolution making it possible to obtain a more accurate solution of the PDEs systems. A seminal work in this subject is the adaptive finite difference method proposed by Berger and Olinger in [4], for hyperbolic problems on rectangular subgrids with arbitrary direction. In the context of numerical weather modelling the AMR method was originally introduced by Skamarock et. al. in [29]. A complete review of this technique and its applications to atmospheric modeling can be found in [20, 25]. This approach has been also used in chemical transport modeling, for revealing the new features of plume concentration profiles [32], in the solution of a shallow water model [23] and in the OMEGA model with a further verification for multiscale simulation of hurricanes [1]. It has been tested for spherical geometries using 2D-AMR based on block-structured grids [20, 21].

Multiresolution methods (MR) appear with the unification of wavelet theory [6]. The wavelet framework became attractive for the adaptive solution of PDEs owing to the localization properties of wavelet functions and the ability of multiresolution representation. The MR appear combined in two types of approaches. In the first one, the wavelet functions are used to locate and refine areas of interest in the discretization process of the domain, while the PDEs system is solved by a standard discretization method (finite differences or finite element method among others). By means of a thresholding process, the wavelet coefficients are used as a local regularity measure in the multiresolution representation of the grid data. In other words, in the context of MR, the adaptive mesh refinement is achieved retaining only the significant wavelet coefficients in the representation of the grid data. In this direction, the sparse point representation method (SPR) developed by Holmström [17-19], is a method widely used in the adaptive solution of hyperbolic equations [12]. In a second approach, the MR are used taking the wavelet functions as basis (trial functions for the Galerkin method [11]) to represent the solution of the PDEs system. This approach has been recently applied to atmospheric and ocean modeling [14, 22].

In this contribution, we make a proposal to improve the methods used to solve numerically the weather forecasting models in the Meteorological Institute in Cuba. Our idea is to consider finite difference schemes over dynamical adaptive sparse meshes, reducing in this way the computing time spent by modeling meteorological events taking place only in small regions of the domain. The sparsity of the meshes is achieved using the SPR method, which is based on an interpolating subdivision scheme [18-19] and a refinement criterion given by a process of thresholding wavelet coefficients. To prove the consistence of the proposal we consider a study case for the shallow water equations. This model is frequently used in the literature to evaluate the performance of new algorithms, see for instance [14, 23-24]. To the best of our knowledge our approach also constitutes a new way to solve such equations. The paper is organized as follows. Section 2 contains a brief review of the SPR method. We also describe the shallow water model and its finite difference discretization on a uniform and SPR refined mesh. In Section 3 we present some experiments and discuss the performance of our proposal comparing with results of the literature. Some conclusion and remarks are finally given.

## 2. SPARSE POINT REPRESENTATION BASED FINITE DIFFERENCE FOR 2D SHALLOW WATER EQUATIONS

In Section 2.1 we describe the SPR method, the underlying interpolating subdivision scheme and its connection with wavelet. In Section 2.2 we present the shallow water equations and its discretization by the leapfrog difference scheme. Some comments about the scheme stability are also given. Section 2.3 is devoted to explain how the leapfrog scheme is adapting on the SPR refined mesh.

### 2.1. Sparse Point Representation mesh refinement and connection with wavelet

The SPR method was introduced by Holmström in [17-19]. Using the SPR approach, nonuniform meshes are built in a little bit different way, regarding classical refinement methods. It essentially removes points from the fine grid, in regions where the solution is smooth, and keep points in regions with large variation of the solution. The points are removed taking as indicator of the smoothness the corresponding wavelet coefficients in the Discrete Wavelet Transform. In fact, we keep the data over the fine uniform grid, but the subsequent computation only includes the remaining points.

Additionally, the SPR provide us with a method to reconstruct any point-value on the original fine grid. The basic idea of the method is based on the interpolating subdivision scheme developed in [9, 13]. Given a data sequence  $s_n$ , defined in dyadic points, the interpolating subdivision scheme computes the points in the halfway in-between each of the points of  $s_n$  leading the sequence  $s_{n+1}$ . The algorithm is as follows:

#### *Interpolating Subdivision Algorithm*

Let be  $f \in L^2(I)$ ,  $I \subset \mathbb{R}$  and let us take a dyadic subdivision of  $I$  with points  $x_n^l = \frac{n}{2^l}$ ,  $n \in \mathbb{Z}$ . We can represent  $f$  by the sequence  $\{s_n^l = f(x_n^l)\}$ , with fixed refinement level  $l \in J = \{0, 1, 2, \dots\}$ . If we want to transform  $s_n^l$  into a sequence  $s_n^{l-1}$  with coarser refinement level  $l-1$ ; saving the corresponding difference between both sequences, we can use the Discrete Wavelet Transform [28]. One step of the DWT look as follows,

$$\sum_{n=0}^{2^l-1} s_n^l \phi_n^l(x) = \sum_{n=0}^{2^{l-1}-1} s_n^{l-1} \phi_n^{l-1}(x) + \sum_{n=0}^{2^{l-1}-1} d_n^{l-1} \psi_n^{l-1}(x),$$

where  $d_n^{l-1}$  are called detail coefficients and they contain the difference between the sequences  $s_n^{l-1}$  and  $s_n^l$ ,  $\phi_n^l(x)$  are integer translations and dyadic dilatations of a scale function  $\phi(x) \in L^2(\mathbb{R})$ , and  $\psi_n^{l-1}$  are integer translations and dyadic dilatations of a function  $\psi(x) \in L^2(\mathbb{R})$  called mother wavelet. In this contribution, the DWT is built following a lifting approach [13], through three steps: split, predict and update. The split stage consists in the separation of the sample regarding the subindexes in even  $s_{2n}^l$  and odd  $s_{2n+1}^l$  subsamples. In the prediction stage, a new approximation of the subsample corresponding to odd subindexes  $\hat{s}_{2n+1}^l$  is predicted, applying the interpolating subdivision scheme to even samples, as follows. Given a sequence  $s_n^l$ , for each group of  $N = 2D$  coefficients  $\{s_{n-D+1}^l, \dots, s_n^l, \dots, s_{n+D}^l\}$ :

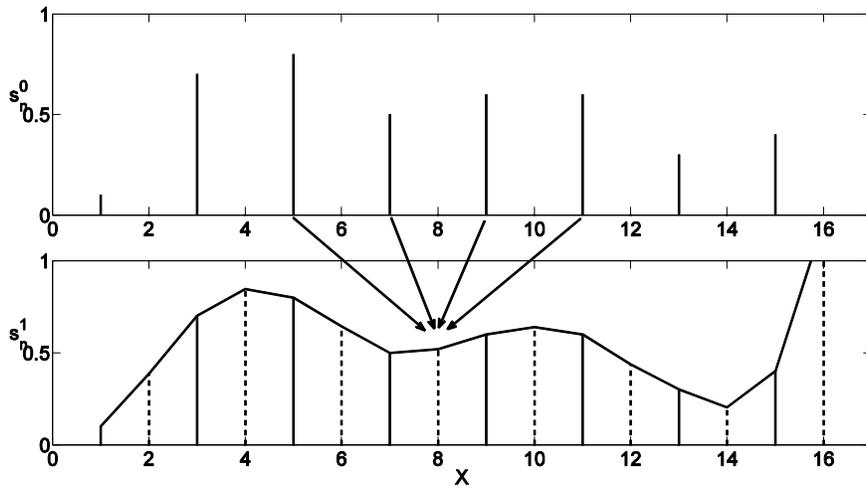
1. Build the polynomial  $p$  of degree  $N - 1$  such that  $p(x_{n+t}^l) = s_{n+t}^l$  for  $-D + 1 \leq t \leq D$
2. Compute the coefficients of the next level as values of  $p(x_{2n+1}^{l+1})$  where  $s_{2n+1}^{l+1} = p(x_{2n+1}^{l+1})$ .

The process to build interpolating subdivision scheme of order  $N = 4$ , are illustrated in Figure 1. For the case of a linear interpolation ( $N = 2$ ), the coefficients of the sequence  $s_n^{l+1}$  are computed according to the formulas:

$$s_{2n}^{l+1} = s_n^l,$$

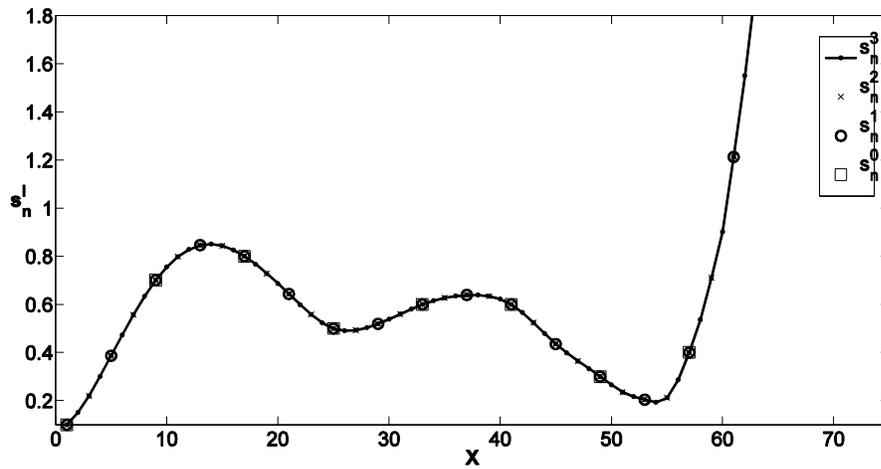
$$s_{2n+1}^{l+1} = \frac{1}{2}(s_n^l + s_{n+1}^l).$$

Notice that the use of polynomials to interpolate the coefficients allows a straightforward adaptation of this scheme to the interval. If for the construction of the polynomial, a point is missing, for instance near the interval's boundaries, then the closest  $p$  points are selected. In Figure 2, we presented three steps of the algorithm for the interpolation scheme of order  $N = 4$ . The initial data sequence with the coarsest resolution is denoted by  $s_n^0$ , while the finest data sequence computed after applying the interpolating subdivision process, is denoted by  $s_n^3$ . A uniform dyadic mesh  $V_l$  is associated to each refinement level  $l$ . Starting with the coarsest mesh  $V_0$  with resolution (spatial step),  $\Delta_x^0$ , the hierarchy of meshes is generated by  $V_0 = \{p = (n\Delta_x^0), 0 \leq n \leq L_x^l\}$ , where  $L_x^l$  is the length of the interval. The Figure 3 shows the associated grids for the one-dimensional case (a) and an example of a two-dimensional case (b).



**Figure 1.** Interpolating subdivision scheme of order  $N=4$ .

Finally, in the update stage, the values of the detail coefficients  $d_n^{l-1}$  are computed as  $d_n^{l-1} = s_{2n+1}^l - \hat{s}_{2n+1}^l$ . The successive application of the discrete wavelet transform over  $s_n^l$  offers a multiscale decomposition into a coarse approximation  $s^0$  and the wavelet coefficients  $d_n^l$  [6]. The wavelet coefficients contain the information related with fluctuations between two consecutive decomposition levels and allow to identify the significance of a point in the sequence, which is the key of the SPR method. The sparse representation considers only the points with wavelet coefficients higher than a given threshold,  $d^l > \sigma$ . Hence, an adaptive mesh is obtained having more points in the areas where the wavelets coefficients exceed  $\sigma$ .



**Figure 2.** Three steps of the interpolating subdivision scheme of order  $N = 4$ .

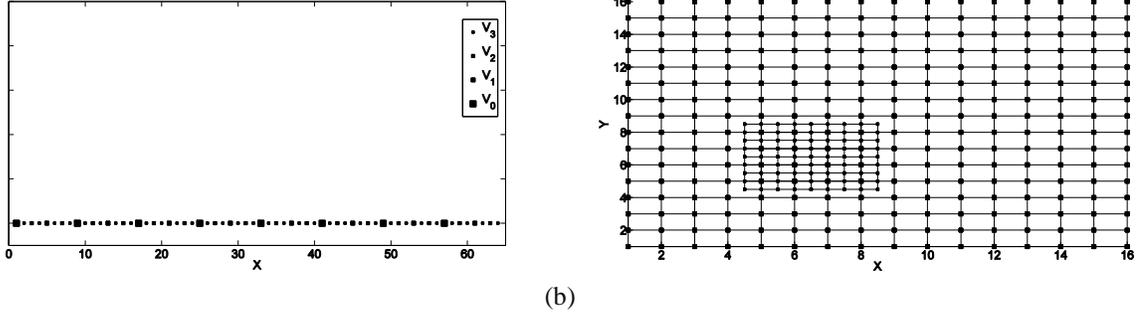


Figure 3. Mesh by the interpolating subdivision scheme: (a) one dimensional and (b) two dimensional.

## 2.2 2D Shallow Water Equations and Finite Difference Discretization

We considered the shallow water model, defined as follows:

$$\frac{\partial u}{\partial t} = (\xi + f)v - \frac{\partial B}{\partial x} + \frac{\tau_x}{\rho_0 h} + A\Delta u - ru, \quad (1)$$

$$\frac{\partial v}{\partial t} = -(\xi + f)u - \frac{\partial B}{\partial y} + A\Delta v - rv, \quad (2)$$

$$\frac{\partial h}{\partial t} = -\frac{\partial}{\partial x}(hu) - \frac{\partial}{\partial y}(hv), \quad (3)$$

where the state and grid variables are the horizontal velocities  $u$  and  $v$  and  $h$  is the surface elevation. The parameter  $B$  is the Bernoulli function, given by  $B = \frac{1}{2}(u^2 + v^2)$ ,  $\xi = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$  is the relative vorticity and  $f$  is the Coriolis parameter. The terms  $A$ ,  $r$  and  $\tau_x$  are the Laplacian diffusivity coefficient, the bottom friction and the wind stress in  $x$ -direction, respectively. The initial conditions for the model are: surface elevation constant and null velocity, with  $A=300$ ,  $r = 0.9e-7$  and  $\tau_0 = 0.05$ .

It is worth pointing out that the shallow water equations are used to describe many physical problems such as tsunami occurrence in oceanic models, studies of cyclonic vortices, atmospheric vortices interaction, among others. The shallow water equations we have used here, were obtained by integration of the Navier-Stokes equations over the depth of the fluid body, by assuming hydrostatic pressure distribution and using the vorticity equation. Several numerical techniques have been successfully used with these models; see for instance [3, 23].

In [7] the model given by Equations (1) - (3) was solved using Finite Difference Schemes on a mesh discretized by adaptive mesh refinement. We propose to solve those equations using Finite Difference Schemes, on a mesh dynamically discretized using the Sparse Point Representation method. Both results are compared to illustrate the consistence of our proposal.

The Finite Difference Method is a well-known and traditional method for the solution of Differential Equations [31], which has been successfully applied in a wide field of researches. The aim is to approximate continuous functions  $u(x, y, t)$  by so-called grid functions  $u(x_i, y_j, t_k)$ , denoting  $u(x_i, y_j, t_k) = u_{ij}^k$ . Let be  $\Delta x > 0$ ,  $\Delta y > 0$ , the fixed grid spacing to discretize the spatial domain and  $\Delta t > 0$ , a fixed step to discretize the time domain. Set  $x_i = i\Delta x$ ,  $y_j = j\Delta y$  and  $t_k = k\Delta t$ ; with  $i, j, k$  integers. The points  $(x_i, y_j, t_k)$  define a regular tridimensional mesh.

Considering the Taylor expansion, derivatives are approximated by difference quotients using the discrete values  $u_{ij}^k$  and so-called difference schemes of different orders are obtained. We use as finite difference scheme the well-known leapfrog formulas [31]. The standard leapfrog scheme for a parabolic equation is conditionally stable. The discretized shallow water equations look as,

$$\begin{aligned} \frac{u_{i,j}^{k+1} - u_{i,j}^{k-1}}{2\Delta t} &= (\xi_{i,j}^k + f_{i,j}^k)v_{i,j}^k - \frac{B_{i+1,j}^k - B_{i-1,j}^k}{2\Delta x} + \frac{\tau_x}{\rho_0 h_{i,j}^k} + A \frac{u_{i+1,j}^k - 2u_{i,j}^k + u_{i-1,j}^k}{\Delta x^2} + A \frac{u_{i,j+1}^k - 2u_{i,j}^k + u_{i,j-1}^k}{\Delta x^2} - ru_{i,j}^k, \\ \frac{v_{i,j}^{k+1} - v_{i,j}^{k-1}}{2\Delta t} &= -(\xi_{i,j}^k + f_{i,j}^k)u_{i,j}^k - \frac{B_{i,j+1}^k - B_{i,j-1}^k}{2\Delta y} + A \frac{v_{i+1,j}^k - 2v_{i,j}^k + v_{i-1,j}^k}{\Delta x^2} + A \frac{v_{i,j+1}^k - 2v_{i,j}^k + v_{i,j-1}^k}{\Delta x^2} - rv_{i,j}^k, \\ \frac{h_{i,j}^{k+1} - h_{i,j}^{k-1}}{2\Delta t} &= -\frac{h_{i+1,j}^k u_{i+1,j}^k - h_{i-1,j}^k u_{i-1,j}^k}{2\Delta x} - \frac{h_{i,j+1}^k v_{i,j+1}^k - h_{i,j-1}^k v_{i,j-1}^k}{2\Delta y}. \end{aligned}$$

As one can observe the equations above can be solved in an explicit way step by step, and from each equation we obtain the unknown variables  $u$ ,  $v$  and  $h$  respectively. Using the Taylor expansion conveniently it can be shown that this leapfrog difference scheme is a consistent scheme of order  $O(\Delta t^2) + O(\Delta x^2)$ , for more details see [31].

Taking into account that the resulting scheme is quite different to what is normally appear in the literature; we explain some details for the stability analysis. We deal with some complexities: we have a system, not a simple one equation, we have partial differential equations, and we have two dimensions. Considering that the third equation can be solve independently we focus our attention in the two first equations. Rewriting equations (1) and (2) in matrix form, without taking into account the term  $\frac{\tau_x}{\rho_0^k i_j^k}$ , because it does not contain the unknown functions,  $u, v$ , we have

$$\mathbf{u}_t = B_0 \mathbf{u}_{xx} + B_1 \mathbf{u}_{yy} + A_1 \mathbf{u}_x + A_2 \mathbf{u}_y + C_0 \mathbf{u}$$

$$B_0 = B_1 = \begin{bmatrix} A & 0 \\ 0 & A \end{bmatrix}, \quad A_1 = \begin{bmatrix} -1 & -1 \\ 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 \\ -1 & -1 \end{bmatrix}, \quad C_0 = \begin{bmatrix} -r & (\xi + f) \\ -(\xi + f) & -r \end{bmatrix}.$$

The bold font  $\mathbf{u}$  denotes a vector of components  $(u, v)$ . The subindexes denote derivatives regarding the

corresponding variable. Using the notation  $\mathbf{u}_{ij}^{k+1} = \begin{bmatrix} u_{ij}^{k+1} \\ v_{ij}^{k+1} \end{bmatrix}$ , the finite difference discretization with leapfrog scheme

can be written as

$$\mathbf{u}_{ij}^{k+1} = \mathbf{u}_{ij}^{k-1} + M_x B_0 \delta_x^2 \mathbf{u}_{ij}^k + M_y B_1 \delta_y^2 \mathbf{u}_{ij}^k + R_x A_1 \delta_x \mathbf{u}_{ij}^k + R_y A_2 \delta_y \mathbf{u}_{ij}^k + 2\Delta t C_0 \mathbf{u}_{ij}^k \quad (4)$$

where  $M_x = \frac{2\Delta t}{\Delta x^2}$ ,  $M_y = \frac{2\Delta t}{\Delta y^2}$ ,  $R_x = \frac{\Delta t}{\Delta x}$ ,  $R_y = \frac{\Delta t}{\Delta y}$ . To investigate the stability the last term in (4) can be neglected (see Theorem 6.2.6 in <sup>[31]</sup>). As you can see in equation (4), we have a three level scheme; this must be reduced to a two level one. With a change of variables

$$\mathbf{u}_{1ij}^{k+1} = \mathbf{u}_{ij}^{k+1}, \quad \mathbf{u}_{2ij}^{k+1} = \mathbf{u}_{ij}^k,$$

$$\mathbf{u}_{1ij}^k = \mathbf{u}_{ij}^k, \quad \mathbf{u}_{2ij}^k = \mathbf{u}_{ij}^{k-1},$$

a two level scheme is obtained,

$$\mathbf{u}_{1ij}^{k+1} = \mathbf{u}_{2ij}^k + M_x B_0 \delta_x^2 \mathbf{u}_{1ij}^k + M_y B_1 \delta_y^2 \mathbf{u}_{1ij}^k + R_x A_1 \delta_x \mathbf{u}_{1ij}^k + R_y A_2 \delta_y \mathbf{u}_{1ij}^k$$

$$\mathbf{u}_{2ij}^{k+1} = \mathbf{u}_{1ij}^k.$$

In matrix form  $\mathbf{U}_{ij}^{k+1} = Q \mathbf{U}_{ij}^k$ ,

$$\text{with } Q = \begin{bmatrix} M_x B_0 \delta_x^2 + M_y B_1 \delta_y^2 + R_x A_1 \delta_x + R_y A_2 \delta_y & I_{2 \times 2} \\ I_{2 \times 2} & \theta_{2 \times 2} \end{bmatrix} \text{ and } \mathbf{U}_{ij}^{k+1} = \begin{bmatrix} \mathbf{u}_{1ij}^{k+1} \\ \mathbf{u}_{2ij}^{k+1} \end{bmatrix}.$$

$I_{2 \times 2}$  and  $\theta_{2 \times 2}$  refers to the identity and zero matrix of order two respectively.

Applying the Discrete Fourier Transform we get  $\hat{\mathbf{U}}^{k+1}(\xi, \eta) = G(\xi, \eta) \hat{\mathbf{U}}^k$  where  $G(\xi, \eta)$  is the so called amplification matrix given by

$$G(\xi, \eta) = \begin{bmatrix} M_x \left(-4 \sin^2 \frac{\xi}{2}\right) B_0 + M_y \left(-4 \sin^2 \frac{\eta}{2}\right) B_1 + R_x (2i \sin \xi) A_1 + R_y (2i \sin \eta) A_2 & I_{2 \times 2} \\ I_{2 \times 2} & \theta_{2 \times 2} \end{bmatrix}.$$

This matrix, as its name suggest, tell us how much the solution errors increase passing from level  $k$  to  $k + 1$ . In order to assure the stability, the spectral radio of  $G(\xi, \eta)$  must be bounded. For normal matrixes, the spectral radio can easily be computed. Then, considering that  $B_0 = B_1 = A I_{2 \times 2}$  we can write the first element of  $G(\xi, \eta)$ ,  $G_{11}(\xi, \eta)$  as:

$$G_{11}(\xi, \eta) = A M_x \left(-4 \sin^2 \frac{\xi}{2}\right) I_{2 \times 2} + A M_y \left(-4 \sin^2 \frac{\eta}{2}\right) I_{2 \times 2} + R_x (2i \sin \xi) A_1 + R_y (2i \sin \eta) A_2.$$

Due to matrixes  $A_1, A_2$  are simultaneously diagonalizable, can be proved (see <sup>[31]</sup>) that exists a matrix  $S$  such that

$$S G_{11} S^{-1} = \begin{pmatrix} -4 A M_x \sin^2 \frac{\xi}{2} - 4 A M_y \sin^2 \frac{\eta}{2} & 0 \\ 0 & -4 A M_x \sin^2 \frac{\xi}{2} - 4 A M_y \sin^2 \frac{\eta}{2} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & -2i R_x (\sin \xi) \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & -2i R_y (\sin \eta) \end{pmatrix}.$$

Hence we can write

$$H = S G(\xi, \eta) S^{-1} = \begin{bmatrix} S G_{11} S^{-1} & I_{2 \times 2} \\ I_{2 \times 2} & \theta_{2 \times 2} \end{bmatrix}$$

and we have that

$$\|G^k\| \leq \|S\| \|S^{-1}\| \|H^k\|.$$

By restricting  $M_x, M_y, R_x, R_y$  from growing, so that the diagonal elements of  $H$  will be less than or equal one, we will have stability. Note that the expressions for  $M_x, M_y, R_x, R_y$  are given by relations between the temporal and spatial steps. Hence (4) is conditionally stable.

## 2.2. Modified leapfrog scheme for the Sparse Point Representation

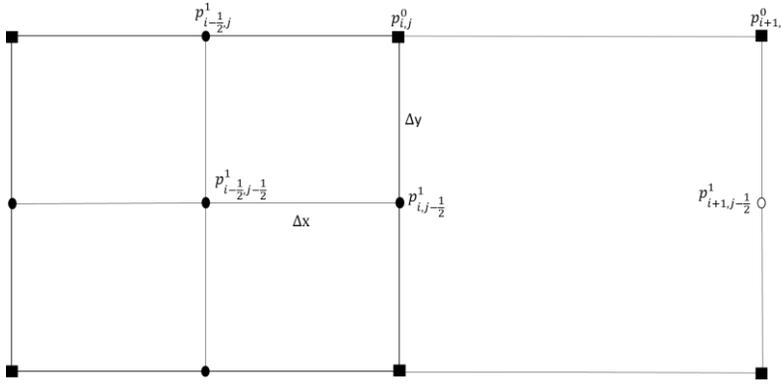
The discretization of the partial derivatives over the sparse grid is based on the idea defined by Holmström in <sup>[17]</sup>, where adaptive spatial steps are used. In this section we present the modified leapfrog scheme on a sparse grid. In Figure 4 a two dimensional sparse grid  $M^l$ , with two refinement levels,  $l = 0, 1$ , related with different spatial steps, is presented.

Let be  $p_x^l \in M^l$ , a point corresponding to the refinement level  $l$ . We denote by  $np$  the minimum number of times that the spatial step of the level of higher refinement  $\Delta x^{l_{max}}$  and  $\Delta y^{l_{max}}$  is included in the distance to the nearest point, in each direction, that is

$$np_x(p_x^l) = \frac{\min\{|x - \bar{x}|, \gamma = (\bar{x}, y) \in M^l, x \neq \bar{x}\}}{\Delta x^{l_{max}}},$$

$$np_y(p_y^l) = \frac{\min\{|y - \bar{y}|, \gamma = (x, \bar{y}) \in M^l, y \neq \bar{y}\}}{\Delta y^{l_{max}}}.$$

Partial derivatives are computed using  $\Delta x = np_x \Delta x$ ,  $\Delta y = np_y \Delta y$ .



**Figure 4.** Sparse mesh  $M$  with two refinement levels. Square points belong to the coarsest level  $M^0$ , black circles belong to the finest level  $M^1$  and the white circles must be obtained by interpolation.

The modified leapfrog scheme of (3), corresponding to the discretization of surface elevation  $h$ , looks as:

$$h_{i,j}^{k+1} = h_{i,j}^{k-1} + 2\Delta t \left( - \frac{h_{i+1,j}^k u_{i+1,j}^k - h_{i-1,j}^k u_{i-1,j}^k}{np_i^k (2\Delta x)} - \frac{h_{i,j+1}^k v_{i,j+1}^k - h_{i,j-1}^k v_{i,j-1}^k}{np_j^k (2\Delta y)} \right). \quad (5)$$

If the value  $h_{i,j}^{k+1}$  coincides with the position of  $p_{i,j}^0$  in Figure 4, then in equation (5),  $np_i^k = np_x(p_x^0) = 1$ .

However for  $h_{i,j}^{k+1} = p_{i+1,j}^0$ , then  $np_i^k = np_x(p_x^0) = 2$ . For  $p_{i,j-\frac{1}{2}}^1$ , we have  $np_i^k = np_x(p_x^0) = 1$ , which means that we need the value in  $p_{i+1,j-\frac{1}{2}}^1$  to compute the derivative. Nevertheless, this point was removed according the sparse representation. In this situation  $p_{i+1,j-\frac{1}{2}}^1$  is recovered by mean of the interpolating subdivision scheme.

The pseudo-code of the proposed algorithm for one of the unknown functions and one time step is as follows:

| Pseudo-code |   |
|-------------|---|
| 1.          | Let $u_{n,m}^{k,l}$ be the sequence of values of $u$ over a 2D mesh with the highest refinement level $l$ .     |
| 2.          | Apply one step of a DWT, first by rows and then by columns  |
| 3.          | Apply thresholding process to wavelet coefficients $d_{n,m}^{(1)l-1}$ , $d_{n,m}^{(2)l-1}$ , $d_{n,m}^{(3)l-1}$ |
| a.          | if $d_{n,m}^{(1:3)l-1} > \sigma$  |
|             | retain the corresponding mesh point   |
| else        | remove it   |

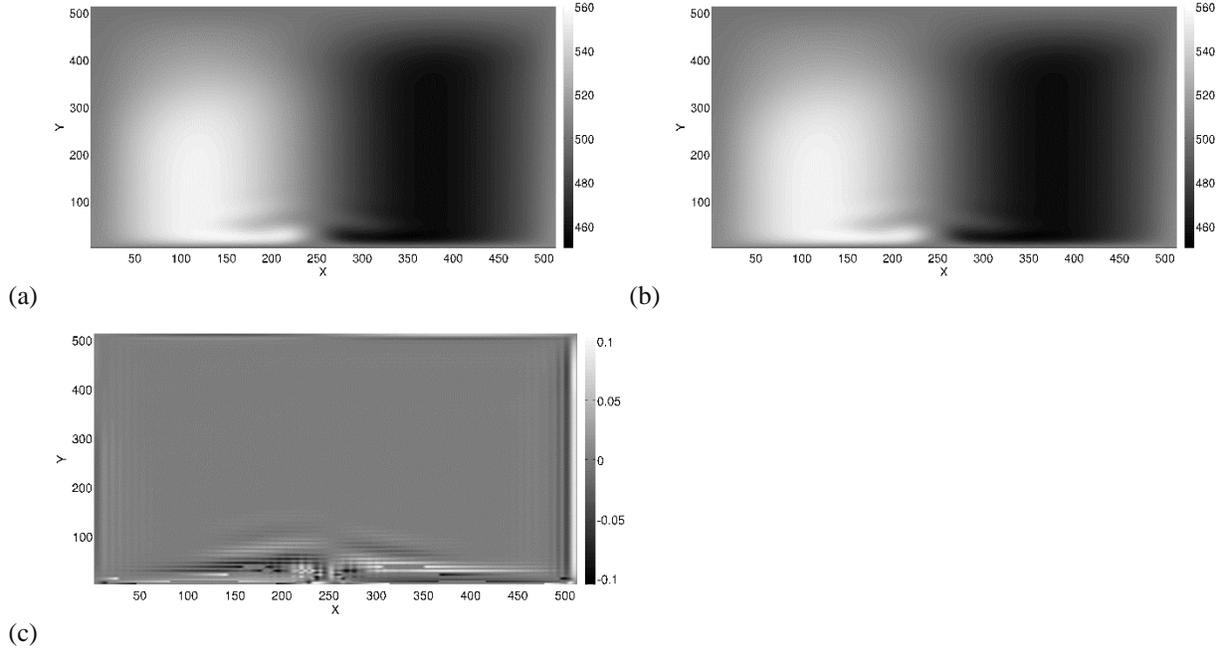
**end if**

4. Repeat 2 and 3 until  $l = \bar{l}$  (coarsest refinement level fixed by the user)
5. Compute  $u_{n,m}^{k+1,l}$  using finite differences over the sparse mesh

### 3. EXPERIMENTS AND RESULTS DISCUSSION

To illustrate the consistence of our approach the shallow water equations were solved by the modified leapfrog scheme on a SPR refined mesh. A total amount of 12 experiments was carried out as follows: the approximate solution of the shallow water equations was computed for three different initial mesh sizes (in term of points):  $129 \times 129$ ,  $257 \times 257$  and  $513 \times 513$ . For each one, we computed the finite difference solution on a uniform and SPR refine mesh. Three different refinement thresholds for the wavelet coefficients were considered by applying SPR method. The interpolating subdivision scheme was of order 4. The physical domain size was  $2580 \text{ Km} \times 2580 \text{ Km}$  and the simulations were conducted in a mesh with initial cells of  $20 \text{ Km}$  size. In other words the initial spatial step is  $\Delta x = \Delta y = 20 \text{ Km}$  (the coarsest spatial resolution). The initial time step is  $\Delta t = 1800 \text{ s}$ .

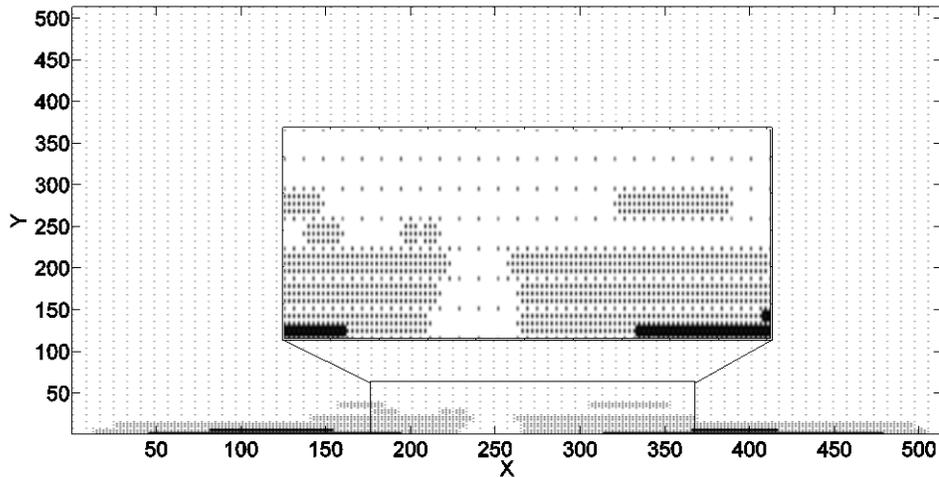
Results were compared with the numerical simulation on a uniform mesh and with a solution where the classic adaptive mesh refinement of Berger and Olinger [4] is used. The implementation used is the so called Adaptive Grid Refinement in FORTRAN (AGRIF), a deep description can be founded in [8]. Two configurations of AGRIF were used. The first one, with a coarse grid of  $129 \times 129$  points and the second with a coarse grid of  $257 \times 257$  points. The refinement process was setting for two and one levels of refinement respectively. The spatial and temporal ratio was setting as 2 and no fixed meshes were used.



**Figure 5.** Results for the surface elevation  $h$ , at  $t=50000$ , initial mesh size  $513 \times 513$ , wavelet threshold  $\sigma = 10^{-1}$  and subdivision scheme order 4; (a) the solution in the uniform grid, (b) the solution using the SPR method, (c) error.

In Figure 5 approximations of the surface elevation  $h$ , for  $t=100000$  in a uniform grid (a) and in the SPR based adaptive grid (b) are shown. Initial mesh size was  $513 \times 513$  and wavelet threshold  $\sigma = 10^{-1}$ . The comparison between the results is possible by recovering the missing values in the sparse representation using the interpolating subdivision scheme. Notice that solutions are similar, in both cases the surface elevation undergo a high variation around the area enclosed by  $0 < y < 100$ ;  $200 < x < 300$ . In Figure 5 (c) the error of the adaptive solution regarding the traditional finite difference solution is shown. Remarkable is that the error order in the solution and the

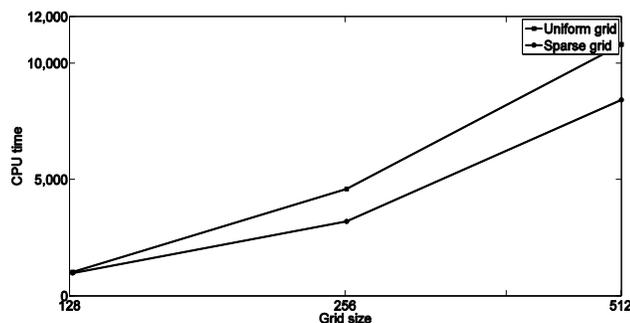
order of the refinement threshold coincide. This has sense since that it is expecting that the loss in accuracy must be in the same magnitude order that the used threshold, as was indicated in [16-17]. In Figure 6 the SPR adaptive mesh is presented. As you can see more points are located in the regions with strong variations. The structure of the mesh is according the behavior of the solution and highlights the ability of the wavelets coefficients as regularity indicator.



**Figure 6.** Adaptive mesh structure. Inside the square box, a zoom of the region limited by  $0 < y < 75$ ;  $175 < x < 375$ .

Fixing the refinement threshold  $\sigma = 10^{-1}$ , we get the behavior of CPU time according grid size variation, as presented in Figure 7. For a mesh size of  $129 \times 129$  points, the standard method is slightly faster than the SPR alternative. This is a consequence of the computational overhead added by calculating the adaptive solution, related with the comparison and interpolating operations. To get a reduction of computational effort applying the SPR method, is crucial to obtain an adaptive mesh with an amount of points very small regarding the original one. For smaller grid size, the points removed from the mesh are not enough to balance the extra calculus to be done. On the other hand, CPU time is significantly decreasing using the SPR mesh refinement, when the mesh size is larger. Here an appropriate balance is reached, between the additional operations of the SPR and the points retained by the refinement criteria. This result indicates that for small grid size, our approach is not recommendable. Analogous results were achieved in [17].

The threshold selection is an important aspect that needs special attention when the SPR method is used. It has a strong relation with the computational cost and the error behavior of the adaptive approach. In Table 1 we present the error variation as well as the compression, regarding thresholds. Here compression means amount of points discarded by SPR algorithm and is computed by



**Figure 7.** CPU time behavior (in seconds), according grid size variation

$$C = \frac{100 * N_{pd}}{N_p},$$

where  $N_p$  is the number of points of the uniform grid with the smallest spatial step and  $N_{pd}$  is the number of discarded points. It can be noted that for large threshold  $\sigma = 10^{-1}$ , less number of points accomplished the refinement condition, which means that few points are retained. This implies less computational effort, but also less accuracy of the results. Indeed, the approximation error is the highest. On the contrary for lesser threshold, the computational cost is high but the accuracy in the results is also high. The compression decrease considerably. The previous analysis lead to another important conclusion, the performance of the SPR is very sensible to threshold selection.

In <sup>[27]</sup> was concluded that for a successful adaptive strategy, compression must be greater than 80 %. There the shallow water equations are solved by an adaptive collocation wavelet method and in all cases, the least compression attained was always greater than 94.6 %. Otherwise, the wavelet method overweight the computational cost. In our case, the highest compression was achieved with  $\sigma = 10^{-1}$ , and it was greater than 92.7 %, for grid sizes 257x257 and 513x513. On the other hand, for grid size 129x129, a very low compression was obtained, consequently, the standard finite difference method is better in this case.

**Table 1.** Compression and accuracy achieved according to mesh size variation and to threshold selection.

| Grid size | $\sigma$  | Points    |          | Error  | Compression % |
|-----------|-----------|-----------|----------|--------|---------------|
|           |           | Discarded | Retained |        |               |
| 129x129   | $10^{-1}$ | 11668     | 4973     | 0.1039 | 70.1          |
|           | $10^{-2}$ | 8695      | 7946     | 0.0621 | 52.5          |
|           | $10^{-3}$ | 8174      | 8467     | 0.0240 | 49.1          |
| 257x257   | $10^{-1}$ | 61178     | 4871     | 0.2115 | 92.6          |
|           | $10^{-2}$ | 58786     | 7263     | 0.0613 | 89.0          |
|           | $10^{-3}$ | 51874     | 14175    | 0.0053 | 78.5          |
| 513x513   | $10^{-1}$ | 255712    | 7457     | 0.1063 | 97.1          |
|           | $10^{-2}$ | 243499    | 19670    | 0.0780 | 92.5          |
|           | $10^{-3}$ | 237910    | 25259    | 0.0061 | 90.4          |

The evaluation of the SPR method against the AMR-AGRIF variants is presented in Table 2. The maximum solution error and CPU time for different time steps were computed. Note that the SPR method slightly improve the accuracy in comparison with the two AMR configurations. In terms of execution time the AMR is better. These results are related with the overhead of SPR in the sense of computational complexity and large memory requirements. What happen to us is also referenced recently in <sup>[10]</sup>, in a comparison between MR and AMR. It is important to remark that our final goal is to incorporate the SPR refinement to an existing numerical weather prediction system, because of that some restrictions must be observed. At the same time, we need by incorporating the SPR method, to do as less as possible code modifications. Even knowing that a block approach for SPR could improve its performance, we cannot use it because of the extra needed effort including in some cases, the complete implementation of the model.

**Table 2.** Comparison between the SPR and two AMR configurations, with one and two levels of refinement.

| Adaptive method | Time ( $10^4$ ) | Maximum error | CPU time (seconds) |
|-----------------|-----------------|---------------|--------------------|
| SPR             | 2.5             | 0.1175        | 1098               |
|                 | 5.0             | 0.1056        | 2286               |
|                 | 7.5             | 0.1045        | 4532               |
|                 | 10              | 0.1063        | 8421               |
| AMR (one level) | 2.5             | 0.0462        | 1043               |
|                 | 5.0             | 0.1079        | 2038               |

|                         |     |        |      |
|-------------------------|-----|--------|------|
|                         | 7.5 | 0.1909 | 4136 |
|                         | 10  | 0.2731 | 8013 |
| <b>AMR (two levels)</b> | 2.5 | 0.1389 | 967  |
|                         | 5.0 | 0.3144 | 1972 |
|                         | 7.5 | 0.5473 | 3855 |
|                         | 10  | 0.8112 | 7809 |

#### 4. CONCLUSIONS

In this work an adaptive method based on sparse representation by means of wavelet coefficients thresholding is presented. A modified leapfrog scheme using Sparse Point Representation mesh refinement was tested by solving a Shallow Water model. Results were compared with the standard leapfrog scheme and with the adaptive mesh refinement technique implemented in the AGRIF code.

The results of our proposal are consistent with those obtained for a uniform grid. The loss of accuracy was of the same order as the used threshold. The better performance of our approach was achieved for larger initial grid size, namely 257x257 and 513x513 and wavelet threshold  $\sigma = 10^{-1}$ , leading a compression > 92.7 %. We confirmed the importance of the threshold selection and its influence on the computational cost. For grid size 129x129; even if a large threshold is selected, the achieved compression through SPR is not sufficient for outcomes the extra calculus needed.

FDS combined with SPR mesh refinement allows more accurate results compared with the AMR-AGRIF approach, but is slightly slower. To our knowledge there are not previous works comparing this two approaches. The improvement obtained by the SPR opens a possibility to enhance operative weather forecasting systems. The computational cost reduction is remarkable without many code changes.

**ACKNOWLEDGEMENTS :** Authors thanks to Prof. Dr. A. Mesejo from the University of Havana, for the corrections and advices and; to Prof. Dr. MO. Domingues from LAC/CTE of the Instituto Nacional de Pesquisas Espaciais in Brazil, for her instruction, support and help. We also thanks the referees for valuable suggestions.

**RECEIVED: FEBRUARY, 2017**  
**REVISED: AUGUST, 2017**

#### REFERENCES

- [1] BACON, D., AHMAD, N.N., BOYBEYI, Z., DUNN, T.J., HALL, M.S., LEE, P.C.S., SARMA, R.A. and TURNER, M.D. (2000): A dynamically adapting weather and dispersion model: The Operational Multiscale Environment Model with Grid Adaptivity (OMEGA). **Mon. Weather Rev.**, 128, 2044–2076.
- [2] BACON, D.P. (2003): Dynamically adapting unstructured triangular grids: A new paradigm for geophysical fluid dynamics modeling. **Proc. of Indian Academy of Science**.
- [3] BARROS, S. R. M. & GARCIA, C. I. (2004): A global semi-implicit semi-Lagrangian shallow-water model on locally refined grids. **Mon. Weather Rev.**, 132, 53–65.
- [4] BERGER, M.J, OLIGER, J. (1984): Adaptive mesh refinement for hyperbolic partial differential equations. **Journal of Computational Physics**, 53, 484–512.
- [5] CENTELLA, A., TAYLOR M.A., BEZANILLA-MORLOT A., MARTINEZ-CASTRO D., CAMPBELL J.D., STEPHENSON T.S. and VICHOT A. (2015): Assessing the effect of domain size over the Caribbean region using the PRECIS regional climate model. **Climate Dynamics**, 44, 1901-1918.
- [6] COHEN, A., (2003): **Numerical Analysis of Wavelet Method**. Elsevier, Amsterdam.
- [7] COLELLA, P., BELL, J., KEEN, N., LIGOCCI, T., LIJEWSKI, M. and VAN STRAALLEN, B. (2007): Performance and scaling of locally-structured grid methods for partial differential equations. **J. Phys.: Conf. Ser.**, 78.
- [8] DEBREU, L., VOULAND, CH. and BLAYO, E. (2008): AGRIF: Adaptive grid refinement in FORTRAN, **Computers & Geosciences** 34, 8-13.

- [9] DESLAURIERS, G. and DUBUC, S. (1989): Symmetric iterative interpolation processes. **Constr. Approx.**, 5, 49-68.
- [10] DEITERDING, R., DOMINGUES, MO., GOMES and S. (2016): Comparison of Adaptive Multiresolution and Adaptive Mesh Refinement Applied to Simulations of the Compressible Euler Equations, **Siam J. Sci. Comput** 38 (5) 173–193.
- [11] DOMINGUES, MO. (2001): Análise Wavelet na Simulação Numérica de Ecuaciones Diferenciales Parciais com Adaptabilidade Espacial. **PhD Thesis**, Departamento de Matemática Aplicada, Universidade Estadual de Campinas.
- [12] DOMINGUES, MO., GOMES SM., DIAZ, LMA. (2003): Adaptive wavelet representation and differentiation on block-structured grids. **Applied Numerical Mathematics**, 47, 421–437.
- [13] DONOHO, D. L. (1992): Interpolating Wavelet Transforms. **Tech. report 408**, Department of Statistics, Stanford University, Stanford, CA.
- [14] DUBOS, T. and KEVLAHAN, NK-R. (2013): A conservative adaptive wavelet method for the shallow-water equations on staggered grids. **Q. J. R. Meteorol. Soc.** 139, 1997-2020.
- [15] FOX-RABINOVITZ, M. S., STENCHIKOV, G. L., SUAREZ, M. J. and TAKACS, L. L. (1997): A finite-difference GCM dynamical core with a variable-resolution stretched grid. **Mon. Weather Rev.** 125, 2943–2968.
- [16] HALEEM, D.A, KESSERWANI, G. and CAVIEDES-VOULLIÈME, D. (2015): Haar wavelet-based adaptive finite volume shallow water solver. **Journal of Hydroinformatics**, 17, 857-873.
- [17] HOLMSTRÖM, M. (1997): Wavelet based methods for time dependent PDEs. **PhD Thesis**, Uppsala University.
- [18] HOLMSTRÖM, M. and WALDEN, J. (1998): Adaptive wavelet methods for hyperbolic PDEs. **J. Sci. Comput.**, 13, 19-49.
- [19] HOLMSTRÖM, M. (1999): Solving hyperbolic PDEs using Interpolating Wavelets, **Siam J. Sci. Comput.**, 21, No. 2, 405–420
- [20] JABLONOWSKI, C. (2004): Adaptive grids in weather and climate modeling. **PhD Thesis**, University of Michigan.
- [21] JABLONOWSKI, C., HERZOG, M., PENNER, J. E., OEHMKE, R. C., STOUT, Q. F., VAN LEER, B. and POWELL, K. G. (2006): Block-structured adaptive grids on the sphere: advection experiments. **Mon. Weather Rev.** 134, 3691–3713.
- [22] KEVLAHAN, N. K.-R., DUBOS, T., and AECHTNER, M. (2015): Adaptive wavelet simulation of global ocean dynamics using a new Brinkman volume penalization, **Geosci. Model Dev.**, 8, 3891–3909.
- [23] LAMBY, P., MULLER, S. and STIRIBA, Y. (2005): Solution of shallow water equations using fully adaptive multiscale schemes, **Int. J. Numer. Methods Fluids** 49(4) 417–437.
- [24] MARTINEZ, EE. and SIERRA-LORENZO, M. (2015): Simulación de un vórtice de velocidades barotrópicas mediante una técnica de refinamiento adaptativo, **Revista Cubana de Meteorología** 21, 18-30.
- [25] NIKIFORAKIS, N. (2005): AMR for global atmospheric modelling. Adaptive Mesh Refinement-Theory and Applications, **Lecture Notes in Computational Science and Engineering**, 41, 505–526, Springer Berlin Heidelberg.
- [26] PINHO, P., DOMINGUES, MO., FERREIRA, P.J.S.G., GOMES, S., GOMIDE, A., and PEREIRA, J.R, (2007): Interpolating Wavelets and Adaptive Finite Difference Schemes for Solving Maxwell's Equations: The Effects of Gridding, **IEEE Transactions on Magnetics**, 43, 1013-1022.
- [27] RECKINGER, Sh.M. and VASILYEV O.V. (2014): Fox-Kemper, Baylor. Adaptive wavelet collocation method on the shallow water model. **Journal of Computational Physics**, 271, 342-359.
- [28] ROUSSEL, O. and ERRERA M.P. (2000): Adaptive mesh refinement: A wavelet point of view, **European Congress on Computational Methods in Applied Sciences and Engineering ECCOMAS 2000, Barcelona** 11-14.
- [29] SKAMAROCK, W. C. and KLEMP, J. B. (1993): Adaptive grid refinements for two-dimensional and three dimensional non-hydrostatic atmospheric flow. **Mon. Weather Rev.**, 121,788–804.
- [30] SKAMAROCK, W.C., KLEMP J.B., DUDHIA, J., GILL, D.O., BARKER, D.M., WANG, W. and POWERS, J. G. (2005): A description of the advanced research WRF Version 2. **National Center for Atmospheric Research Technical Note** NCAR/TN468+STR, Boulder, CO, USA.
- [31] THOMAS, J.W. (2013): **Numerical Partial Differential Equations: Finite Difference Methods**. Springer Science & Business Media.
- [32] TOMLIN, A., BERZINS M., WARE J., SMITH J., and PILLING M. J. (1997): On the use of adaptive gridding methods for modelling chemical transport from multi-scale sources. **Atmospheric Environment**, 31, 2945 – 2959.

- [33] WANG, Y. (2001): An explicit simulation of tropical cyclones with a triply nested movable mesh primitive equation model: TCM3. Part I: Model description and control experiment, **Mon. Wea. Rev.**, 129, 1370–1394.
- [34] XUE, M., DROEGEMEIER, K.K., WONG, V., SHAPIRO, A., and BREWSTER, K. (1995): **ARPS Version 4.0 user's guide**. Center for Analysis and Prediction of Storms, Univ. of Oklahoma, 380.
- [35] ZHANG, D.L., CHANG, H.R., SEAMAN N, L., WARNER, T.T. and FRITSCH, J.M. (1986): A two-way interactive nesting procedure with variable terrain resolution, **Mon. Wea. Rev.**, 114, 1330–1339.