A refining estimation for adaptive solution of wave equation based on curvelets

Jianwei Ma, Gang Tang, M. Hussaini

Jianwei Ma, Gang Tang, M. Y. Hussaini, "A refining estimation for adaptive solution of wave equation based on curvelets," Proc. SPIE 6701, Wavelets XII, 67012J (20 September 2007); doi: 10.1117/12.730336

SPIE.

Event: Optical Engineering + Applications, 2007, San Diego, California, United States

A Refining Estimation for Adaptive Solution of Wave Equation based on Curvelets

Jianwei Ma [1], Gang Tang [1], M. Y. Hussaini [2]

[1.] School of Aerospace, Tsinghua University, Beijing 100084, China
[2.] School of Computational Science, Florida State University, Tallahassee, Florida 32306

ABSTRACT

This paper presents a refining estimation to control the process of adaptive mesh refinement (AMR) based on the curvelet transform. The curvelet is a recently developed geometric multiscale system that could provide optimal approximation to curve-singularity functions and sparse representation of wavefront phenomena. Utilizing these advantages, we attempt to introduce the curvelet transform into AMR as a criterion estimate of refinement for adaptive solving of the wave equation. Numerical simulations show that the proposed method could optimally capture interesting areas where refinement is needed, so that a high accuracy result is obtained.

Keywords: refining estimation, curvelet transform, wave equation, adaptive

1. INTRODUCTION

During the last decade, the wavelet transform has been widely applied in many areas, such as signal processing, data compression, pattern recognition and equation solving, mainly because of its good time-frequency localization ability and best nonlinear approximation ability for one-dimensional piece-wise smooth functions. However, the primary advantage of wavelet analysis is to represent "point singularity" of signals, that is to say, it only can represent locations and features of singular points. But for two or more dimensions, the commonly used two-dimensional wavelet, which is the tensor product of one-dimensional wavelets, is roughly isotropic. It can only represent three directional information and "point singularity" features, but fails to deal with those smooth away from "linear singularity" or "curve singularity". In order to overcome these limitations, a new theory of multiscale geometric analysis, namely, curvelet transformation was developed.

The first generation of the curvelet transform (Curvelet99) was developed by Candès and Donoho in 1999 [1], deriving from ridgelet theory. But its digital implementation is too complex, which needs many steps including subband decomposition, smooth partitioning, renormalization and ridgelet analysis. In addition, its pyramidal decomposition method brings a tremendous amount of data redundancy. To overcome these shortcomings, a second-generation curvelet transform (curvelet2004, hereafter referred to as "curvelet") was developed [2]. Unlike the original curvelet transform, this construction does not use ridgelets. It defines the specific forms of curvelet bases directly in frequency domain, which makes it understood and implemented more easily, thus it should be called "curvelet" in a real sense. Then a fast discrete curvelet transform was developed based on this [3], which prompted various applications in the field of image processing[4-7].

Different from wavelets, curvelets are indexed by an additional parameter called orientation (angle) parameter, besides scale and location parameters, so they have a better orientation recognition ability. Furthermore, the elements in curvelets are highly anisotropic at fine scales, with effective support shaped according to the parabolic scaling principle $width \approx length^2$, so they can well represent the geometric regularity of an image, which makes curvelets have optimal approximation ability for functions with curve singularity and allows curvelets to capture geometric features easily. Hence, we believe that the curvelets could better capture the features of propagating waves in solving hyperbolic equations. Le Demanet *et al.* have done some useful work on the theoretical aspects [8,13,14]. They proved that the curvelet representation of the solution operator $E(t)$ is both optimally sparse and well organized for a wide class of linear hyperbolic differential equations. Firstly they introduced a solution operator of the system, and then worked out the solution of the wave equation by exploring a sparse expression of the operator $E(t)$. This method was primarily shown

Wavelets XII, edited by Dimitri Van De Ville, Vivek K. Goyal, Manos Papadakis,
Proc. of SPIE Vol. 6701, 67012J, (2007) · 0277-786X/07/$18 · doi: 10.1117/12.730336

to be an effective way, but it has to introduce a solution operator first and get a sparse expression for the operator, which is very complex. However, to the best of our knowledge, there are a few reports[13,14] on solving partial differential equations using the curvelet transform yet, besides this. On the other hand, the adaptive mesh refinement (AMR) method [9] has been applied widely in the field of numerical computation, especially in solving wave propagation problems. If we can capture wave fronts effectively through some method, and then do a mesh refinement in the regions, which require higher order accuracy, but use a coarser grid elsewhere to save calculation time. Many researchers have done a lot of work on the AMR, for example, Blayo and Debreu[10] examined the application of AMR for a structured mesh in the context of ocean modeling, Pretorius and Choptuik[11] presented a modification to the B&O AMR method to solve coupled elliptic-hyperbolic systems. However, most of these AMR methods use truncation error (TE) to control the implementing process of adaptive mesh refinement, which needs to judge whether the TE meet the required accuracy in each iteration step.

Since the curvelet transform and AMR method both have such distinct advantages for solving wave propagation problems, we would naturally like to couple them to reinforce the advantages and preclude any disadvantages. In this paper, we propose curvelet-based AMR method for the solution of partial differential equations.

2. CURVELET TRANSFORM

In this paper, we use the second generation curvelet transform to control the grid refinement during the AMR process. We briefly review the curvelets, and refer the reader to [2,3,8] for details.

Curvelet is defined directly in continuous space, whose basis functions $\varphi_\mu(x)$ (μ represents the subscript (j,l,k)) are defined in frequency domain as:

$$\widehat{\varphi}_\mu(\omega) = \widehat{\varphi}_{j,l,k}(\omega) = 2\pi\chi_{j,l}(\omega)u_{j,k}(R_{j,l}(\omega)), \omega = (\omega_1,\omega_2)^T, k = (k_1,k_2)^T, k_1,k_2 \in z. \quad (1)$$

Then the curvelet transform of function f can be expressed as the inner product of f and $\varphi_\mu(x)$:

$$c_\mu = < f, \varphi_\mu > = \int_{R^2} f(x)\overline{\varphi}_\mu(x)dx, \quad (2)$$

where $R_{j,l}$ is the rotation by radians $\theta_{j,l} = 2\pi l 2^{-j}$, and u is a two-dimensional complex harmonic function:

$$u_{j,k}(\omega) = [2^{-3j/2}/(2\pi\sqrt{\delta_1\delta_2})]e^{i(k_1+1/2)2^{-2j}\omega_1/\delta_1}e^{ik_2 2^{-j}\omega_2/\delta_2}, \quad (3)$$

where， $\delta_1 = 14/3(1+O(2^{-j})), \delta_2 = 10\pi/9$. χ is a 2-D window function：

$$\chi_{j,l}(\omega) = w(2^{-2j}|\omega|)(v_{j,l}(\theta)+v_{j,l}(\theta+\pi)). \quad (4)$$

Here $|\omega| = \sqrt{\omega_1^2+\omega_2^2}$， $v_{j,l}(\theta) = v_{j,l}(2^j\theta - \pi l)$ is assumed to be an even, C^∞-angular window that is supported on $[-\pi,\pi]$ and obeys

$$|v^2(\theta)|^2 + |v^2(\theta-\pi)|^2 = 1, \theta \in [0,2\pi), \quad (5)$$

$$\sum_{l=0}^{2^{j+1}-1} |v(2^j\theta - \pi l)|^2 = 1, j \geq 0. \quad (6)$$

The radial window w is a compactly supported function which obeys

$$|w_0(t)|^2 + \sum_{j\geq 0} |w(2^{-2j}t)|^2 = 1, t \in R. \quad (7)$$

Let $\chi_0^2(\omega) = w_0^2(|\omega|) + w^2(|\omega|)$， and construct a suitable radial window function w, making χ obeys

$$\left|\chi_0(\omega)\right|^2 + \sum_{j\geq 1}\sum_{l=0}^{2^{j-1}}\left|\chi_{j,l}(\omega)\right|^2 = 1. \tag{8}$$

The formula (8) shows that χ realized a multi-scale orientation tilling of the frequency domain, dividing the domain into many wedge tilings in different directions and scales, shown as Fig. 1.

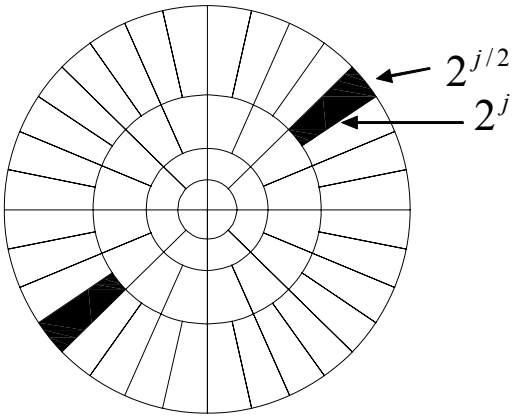

Fig. 1. Curvelet tilling of the frequency plane. In the frequency domain, curvelets are supported near symmetric "parabolic" wedges, which obey approximately the scaling relationship $width \approx length^2$. The shaded area represents such a generic wedge.

A low-pass window W_0 is introduced for the coarsest scale, which obeys

$$\left|W_0(r)\right|^2 + \sum_{j\geq 0}\left|W(2^{-j}r)\right|^2 = 1. \tag{9}$$

Then for $k_1, k_2 \in Z$, define the curvelet in coarse scale

$$\begin{cases} \Phi_{j_0,k}(x) = \Phi_{j_0}(x - 2^{-j_0}k), \\ \hat{\Phi}_{j_0}(\omega) = 2^{-j_0}W_0(2^{-j_0}\left|\omega\right|). \end{cases} \tag{10}$$

It is obvious that curvelet's components in coarse scales do not have directional property. That is to say, the entire curvelet transform is composed of directional components $(\varphi_{j,l,k})_{j\geq j_0,l,k}$ in fine scales and isotropic wavelets $(\Phi_{j_0,k})_k$ in coarse scales. The supported interval of curvelet basis functions obeys an anisotropy scaling relation

$$length \approx 2^{-j/2}, width \approx 2^{-j} \Rightarrow width \approx length^2, \tag{12}$$

which is the core property for curvelets having optimal approximation ability for functions with curve singularity and capturing the geometric features easily.

The curvelet reconstruction formula is

$$f = \sum_{\mu}\left\langle f, \varphi_\mu(x)\right\rangle \varphi_\mu(x). \tag{13}$$

The M -term partial reconstruction f_M^C obtained by selecting the M largest terms in curvelet bases obeys

$$\left\|f - f_M^C\right\|_2^2 \approx O(M^{-2}(\log M)^3). \tag{14}$$

3. THE CURVELET BASED AMR ALGORITHM

If the grid is very coarse, it would of course reduce the computational cost but at the risk of loss of accuracy. If the grid is too fine, it may unnecessarily improve accuracy at possibly prohibitive computational cost. Hence, the development of an effective method that balances the computational and accuracy has become a important research topic [9, 10,11]..

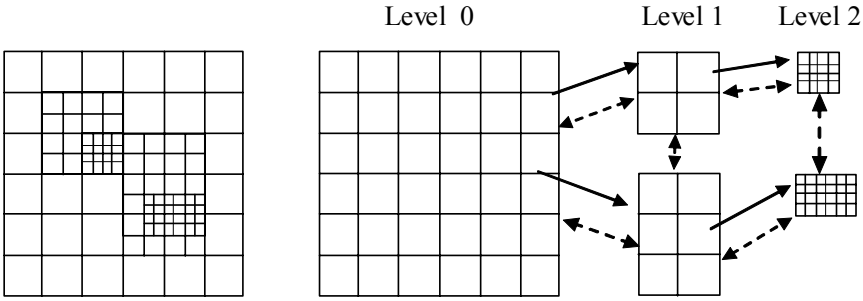

Fig. 2. An example of the grid hierarchy. It consists of three levels, where each level contains a set of uniform meshes of the same resolution. The figure on the left represents the computational domain, covered by a three-level-deep hierarchy. The right figure shows how the hierarchy is stored in memory and how the data are transferred between different levels. The solid-line arrows show the grid relationship between child grids and their parent grids. While the dashed double-way arrows show the data transfer way.

3.1 Mesh refinement method

First generate a coarse grid as the base grid for the whole domain, then refine some local grids where we are more interested, in order to improve the accuracy of these areas. As shown in Fig.2, if its accuracy has not meet the expectation for a certain local area after refined once, we can do a further refinement in this area. Here one crucial problem is to find the criterion for mesh refinement.

Berger et al. [9] used the truncation error to control the local refinement. Let us first give a brief review of their classical work:

The error estimate is obtained by Richardson's method. Let $R_{\Delta h}$ be a finite difference operator based on mesh width Δh and time step Δt. Then we have

$$u(x,t+\Delta t) = R_{\Delta h}u(x,t) + (\Delta h)^{q+1} f(x,t) + (\Delta t)^{q+1} g(x,t) + O(\Delta h^{q+2} + \Delta t^{q+2}), \tag{15}$$

where u is a sufficiently smooth solution of the equation. If we advance two time steps, we have

$$u(x,t+2\Delta t) = R_{\Delta h}^2 u(x,t) + 2(\Delta h)^{q+1} f(x,t) + 2(\Delta t)^{q+1} g(x,t) + O(\Delta h^{q+2} + \Delta t^{q+2}). \tag{16}$$

On the other hand, if we change the mesh width into $2\Delta h$ and $2\Delta t$, also using the same Richardson's difference operator, we will obtain

$$u(x,t+2\Delta t) = R_{2\Delta h}u(x,t) + (2\Delta h)^{q+1} f(x,t) + (2\Delta t)^{q+1} g(x,t) + O(\Delta h^{q+2} + \Delta t^{q+2}). \tag{17}$$

Subtracting (16) from (17), we obtain the local truncation error:

$$(\Delta h)^{q+1} f(x,t) + (\Delta t)^{q+1} g(x,t) = \frac{R_{\Delta h}^2 u(x,t) - R_{2\Delta h}u(x,t)}{2^{q+1} - 2} + O(\Delta h^{q+2} + \Delta t^{q+2}). \tag{18}$$

Then we can use these estimates to determine where to refine the grid. Typical calculations give an optimal regridding frequency of approximately every 3-4 steps. Compared with a threshold given in advance, we perform a mesh refinement in the local grid if the estimate is less than the threshold, or go to next step.

Use the same CFL number for all levels

$$CFL = \frac{\Delta t}{\Delta h}. \tag{19}$$

The refinement ratio of spatial and temporal resolution between two adjacent levels is a given integer r. If $\Delta h_{l+1} = r\Delta h_l$ on the grid level l, there will be $\Delta t_{l+1} = r\Delta t_l$ and the CFL criterion will be automatically satisfied for all grids.

3.2 Curvelet-based AMR

The present method employs the curvelet transform to provide the necessary criterion for adaptive mesh refinement. See Fig. 3 for a flow chart of our method, which will be described step by step in the following subsection.

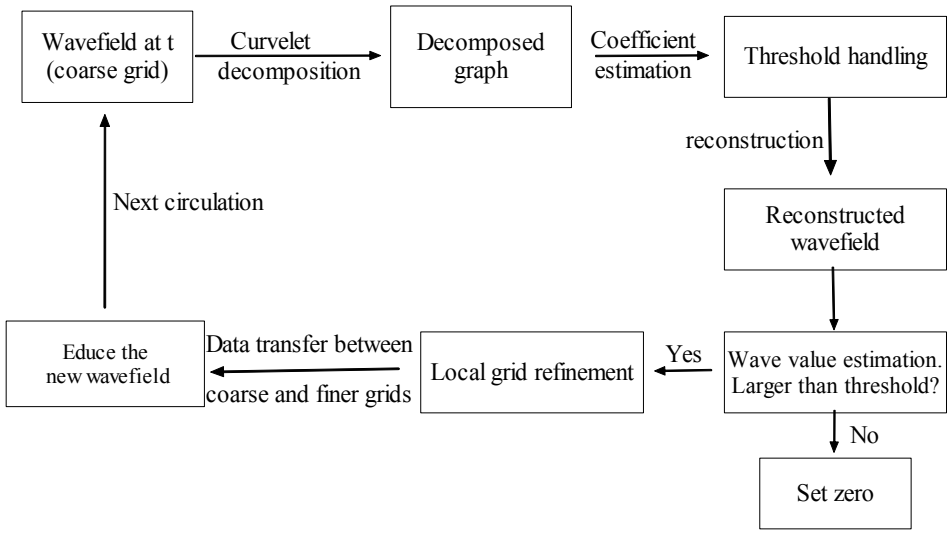

Fig. 3. A flow chart to show the adaptive process of AMR, which uses the refining criterion based on the curvelet transform.

3.2 Curvelet decomposition and reconstruction

As shown in Fig. 3, we perform curvelet decomposition on the current wavefield, and then apply a threshold to reconstruct it. With many advantages such as a directional property, curvelets should give an optimal approximation to the wavefield and capture the wave-front well. Thus, we could filter the small values and highlight the areas which need higher accuracy for local mesh refinement by decomposition and reconstruction using the curvelet transform.

3.3 Estimation for refinement

Through the wavefield reconstruction mentioned above, we could do some estimation on the reconstructed wavefield in order to know which local areas need to be refined. We set a threshold first, then scan the entire wavefield block by block, and do a mesh refinement if the mean wave value of a local area is larger than the threshold.

Furthermore, if we do it as a customary way, we may control the refinement according to the values of each local area's curvelet coefficients. But it is not so easy to find out the corresponding curvelet coefficients sub-matrix of a local area because the data structure of curvelet is not so regular and perfect yet. While if we mesh the entire wavefield into different small areas first, and then do a curvelet transform in every small area separately, the properties of curvelets (e.g. directional property) will not behave so well and couldn't be used effectively. So we chose to do threshold estimation for refinement in the reconstructed wavefield at the moment.

3.4 Data storage and transfer

Using hierarchical storage structure, we first store the base grid points (level 0) in a location, and then store all of the grid points for levels from level 1 to the highest level in separate locations. The data could be transferred between parent grids and sub-grids, and also between the grids within the same level. That is to say, the data could be transferred between the neighboring grid levels, and between the neighboring grids at the same level (See figure 2). The data from different levels are stored separately, so that every level is composed of uniform grids. As its data structure is regular that we can store and mange it easily. The advantages of this storage strategy are apparent, which is prevalent in many fields.

3.5 Local mesh refinement and data transfer

According to the estimation method, we refine the grids at some local areas to obtain higher accuracy at a time step t. We can get the boundary values of an area to be refined, through interpolation from its parent grid points at the former time step $t - n\Delta t$, and then do a local refinement and transfer the new data back to the coarser grid points. In this paper, as we mainly attempt to test the effect of the estimation method using the curvelet transform, for simplicity, we did not strictly adhere to the actual AMR method, but a simplified version. As shown in Fig.4, for the area where we only refine it only once, and after being refined, we let the values of refined grid points equal those of the same finer-level grid that have been computed in advance.

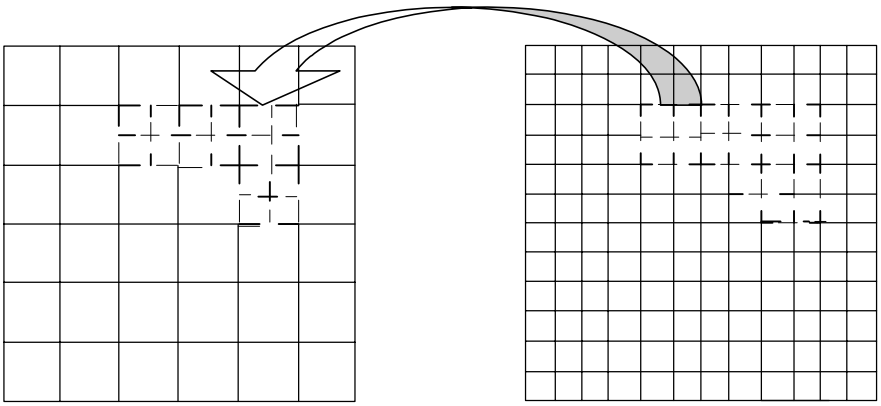

Fig. 4. An example of a simplified way to realize AMR result. The left figure represents a computational domain of AMR, and the right figure shows a corresponding refined mesh of the domain whose points' values have been computed in advance. The dashed-line area and the arrow show the relationship of corresponding points' values assignment.

It will certainly influence the accuracy as we have introduced more precise values. But at least it approximately reflects the capability of our method for locating the wave-front. And we can realize this step easily through AMR in practice.

4. EXPERIMENTS

4.1 Experiment 1

In the first example, we show the validity of the proposed method. Consider a mixed initial boundary value problem for the two-dimensional wave equation

$$\begin{cases} \dfrac{\partial^2 u}{\partial t^2} = \dfrac{\partial^2 u}{\partial x^2} + \dfrac{\partial^2 u}{\partial y^2}, (x,y) \in D, t > 0, \\ u\big|_{t=0} = 0, (x,y) \in D, \\ \left.\dfrac{\partial u}{\partial t}\right|_{t=0} = 5(\sin x + \cos y), (x,y) \in D, \\ u\big|_{\Gamma} = g(x,y,t), (x,y) \in \Gamma, t \geq 0, \end{cases} \tag{20}$$

where $D = \{(x,y) | 0 < x, y < L\}$ is a square domain with a boundary Γ and

$$g(x,y,t) = \begin{cases} 5\cos y \sin t, x = 0, \\ 5(\sin L + \cos y)\sin t, x = L, \\ 5\sin x \sin t, y = 0, \\ 5(\sin x + \cos L)\sin t, y = L. \end{cases} \tag{21}$$

There is an analytic solution [12] $u(x,y,t) = 5(\sin x + \cos y)\sin t$ in domain D for problem (20), and we consider a wavefield in $D = \{(x,y) \mid 0 < x, y < 50\}$, using three types of grid-computing methods to solve for the wavefield at $t = 0 \sim 4s$. The time and space step sizes are: $\Delta t = 0.1$ and $\Delta x = \Delta y = \pi / 14$.

The result is shown in Fig. 5, where the left figure shows the computed wavefield using coarse grid at time $t = 3.9s$, and the right one shows the areas requiring refinement determined by the estimation method based on the curvelet transform.

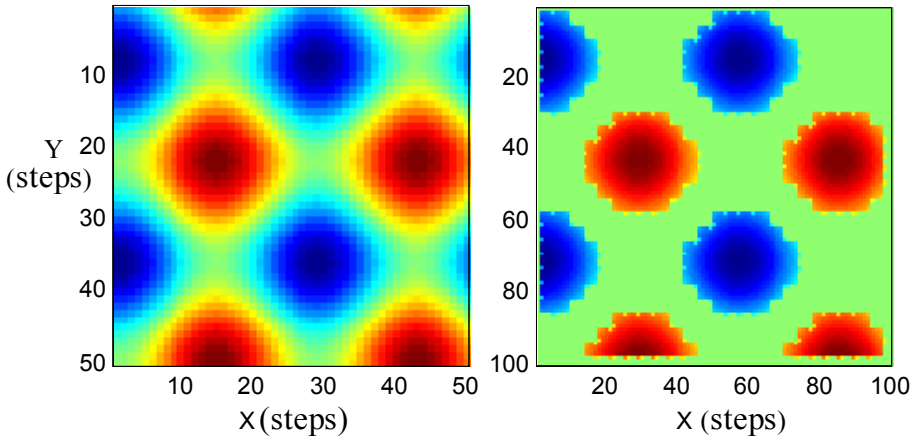

Fig. 5. A wave field snapshot at $t = 3.9s$. The figure on the left shows the snapshot in coarse grid. The right figure shows the captured areas that need to be refined based on determination by the refinement criterion based on the curvelet transform. The right figure has been refined, so its number of step is twice that of the left one, while its step-length is half of the left one.

The mean error with reference to the exact analytical solution is computed using different grids from coarse to fine, and it is plotted as a function of time step in Fig.6. The adaptive gird balances the computational complexity and accuracy.

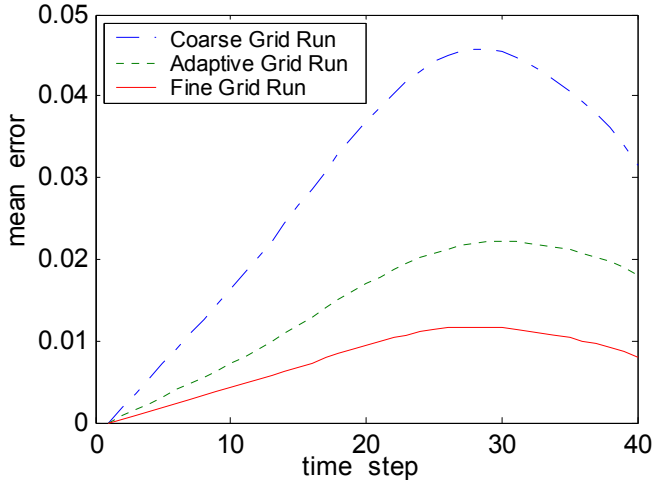

Fig. 6. Error comparisons of the simulations based on different grids.

4.2 Experiment 2

Two-dimensional sound wave equation including hypocenter is given by

$$\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial z^2}+f(t)=\frac{1}{v^2}\frac{\partial^2 u}{\partial t^2} \tag{22}$$

where u denotes the sound wavefield $u(x,z,t)$, v is the sound wave's velocity $v(x,z)$, and $f(t)$ denotes the hypocenter. Performing a 7-point difference scheme to the equation above, we obtain

$$u_{i,j}^{k+1}=2u_{i,j}^{k}-u_{i,j}^{k-1}+A^2(u_{i+1,j}^{k}+u_{i-1,j}^{k}+u_{i,j+1}^{k}+u_{i,j-1}^{k}-4u_{i,j}^{k})+v^2(\Delta t)^2 f(k\Delta t) \tag{23}$$

where, $u_{i,j}^{k}=u(i\Delta x, j\Delta z, k\Delta t)$, $\Delta x, \Delta z,$ and Δt respectively denotes the step length in space and time domains, $A=\frac{v\Delta t}{h}, h=\Delta x=\Delta z$, and $f(t)$ is a hypocenter function given by

$$f(t)=\begin{cases}\sin(50t)\times e^{-188(t-3\pi/100)^2}, 0\le t\le 6\pi/100\\ 0, t>6\pi/100\end{cases} \tag{24}$$

Here we set $h=0.01$, $\Delta t=2ms$, and $v=1$. A snapshot at $t=400ms$ is shown in Fig.7.

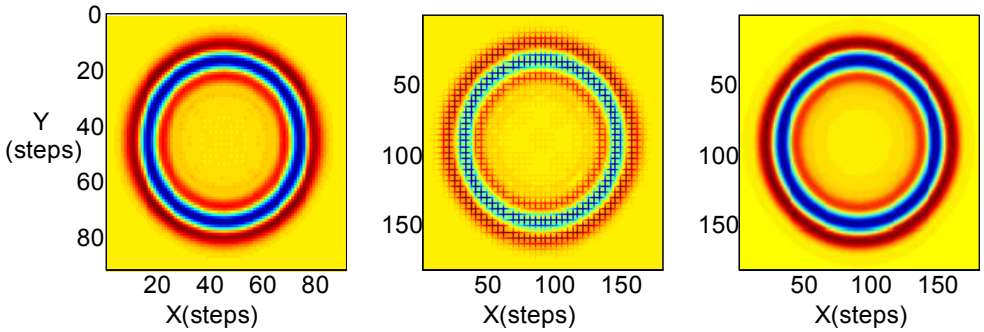

Fig. 7. Snapshots at $t = 400ms$. The left figure is computed on a coarse grid, while the right figure is on a fine grid. The figure in the middle shows the areas where refinement is needed, which is determined by the refinement criterion based on curvelet transform.

In Fig. 8, we display the curvelet coefficients of the wavefied snapshot at $t = 400ms$ computed on coarse grid.

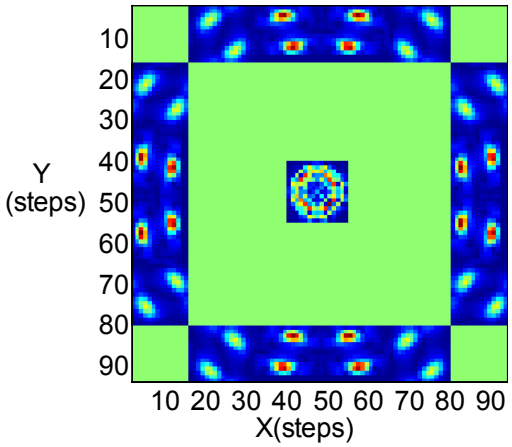

Fig. 8. The log picture of the curvelet coefficients of Fig. 7 (left).

From the above-mentioned tests, we can see that the curvelets can sparsely represent the wavefild only using a few non-zero coefficients, and that these coefficients could optimally approximate the wavefield.

5. CONCLUSIONS

In this paper we presented a refinement criterion for the AMR algorithm based on the curvelet transform. Since the curvelets could optimally approximate to "curve singularity" functions and sparsely represent the wavefront phenomena, we attempted to apply its theoretical insights into an effective algorithm AMR, in order to seek a fast accurate method for simulation of a wavefield. From the experiments, we see that the proposed refinement criterion could effectively capture wavefronts, which provides a useful way to determine where refinement is needed for adaptive solution of the seismic wave equation.

So far we have used a simplified way to realize the refinement during the AMR process, mainly because the disadvantages of the current curvelet version: bad data structure and lack of analytical expressions in space. Thus, improvement of the data structure of curvelet transform and construction of analytical curvelet frames for developing better AMR algorithms are the themes of our future work.

Acknowledgment: The first author would like to thank the financial support from TBRF, PetroChina, and NSFC-10572072.

REFERENCES

1. E. J. Candès and D. L. Donoho, "Curvelets-A surprisingly effective nonadaptive representation for objects with Edges", *in Curves and Surface Fitting: Saint-Malo 1999*, A. Cohen, C. Rabut, L. L. Schumaker (Eds.), Vanderbilt Univ. Press, Nashville, 2000, pp. 105–120.
2. E. J. Candès and D. L. Donoho, "New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities", *Commun. Pure Appl. Math.*, 57 (2), 219-266 (2004).
3. E. J. Candès, L. Demanet, D. L. Donoho, and L. Ying, "Fast discrete curvelet transforms", *Multiscale Model. Simul.*, 5 (3), 861-899 (2006).
4. J. L. Stark, E. J. Candès, and D. L. Donoho, "The curvelet transform for image processing", *IEEE Trans. Image Process*., 11 (6), 670-684 (2002).
5. J. Ma, A. Antoniadis, and F.-X. Le Dimet, "Curvelet-based snake for multiscale detection and tracking of geophysical fluid", *IEEE Trans. Geosci. Remote Sens.*, 44 (12), 3626-3638 (2006).
6. J. Ma and G. Plonka, "Combined curvelet shrinkage and anisotropic nonlinear diffusion", *IEEE Trans. Image Process*., 16 (9), (2007).
7. J. Ma, "Curvelets for surface characterization", *Appl. Phys. Lett.*, 9, 054109 (2007).
8. L. Demanet, "Curvelets, wave atoms, and wave equations", Ph. D thesis, California Institute of Technology, Pasadena, California, May 2006.
9. M. Bergerand and J. Oliger, "Adaptive mesh refinement for hyperbolic partial differential equations", *J. Comput. Phys.*, 53, 484-512 (1984).
10. E. Blayo and L. Debreu, "Adaptive mesh refinement for finite-difference ocean models: first experiments", *J. Phys. Oceanography*, 29, 1239-1250 (1999).
11. F. Pretorius and M. W. Choptuik, "Adaptive mesh refinement for coupled elliptic-hyperbolic systems", *J. Comput. Phys.*, 218, 246-274 (2006).
12. C. Jin, S. Zhang, and X. Ding, "A class of parallel difference methods for solving two-dimensional wave equation", *J. Nature Science of Heilongjiang Univ.*, 17(4), 14-17 (2000) (in Chinese).
13. E. J. Candès and L. Demanet," Curvelets and fourier integral operators", *C.R. Acad. Sci. Paris, Sér. I Math*, 36, 395-398 (2003).
14. E. J. Candès, L. Demanet, "The curvelet representation of wave propagators is optimally sparse", *Comm. Pure Appl. Math.,* 58 (11) 1472-1528, 2005.