Adaptive Mesh Refinement for Finite Difference WENO Schemes

Shengtai Li and J. Mac Hyman

Theoretical Division, MS B284, Los Alamos National Laboratory, Los Alamos, NM 87545

Los Alamos Report LA-UR-03-8927

Abstract

In this paper, a high order finite difference weighted essentially non-oscillatory (WENO) scheme is combined with adaptive mesh refinement (AMR) to solve problems that require multi-resolution in space and time. Because conservative finite difference schemes of third or higher order of accuracy can only be achieved on uniform rectangular or smooth curvilinear meshes, either conservation or high order accuracy is lost when the WENO schemes are applied to the AMR framework. Several implementations are considered and compared to see which is more important: accuracy or conservation. The numerical experiments show that the general conservative implementation of AMR for finite-volume method can be used for high order WENO schemes without loss of much of accuracy.

1 Introduction

The significance and advantages of using high order (higher than three) schemes in numerical simulation have been demonstrated by many applications (see Shu [17]). WENO schemes are designed based on the successful essentially non-oscillatory (ENO) schemes (see [15] and its references). High order finite volume and finite difference WENO schemes have been

Email address: sli@lanl.gov (Shengtai Li and J. Mac Hyman).

constructed in the past decade (see the review paper [17] and its references). It is well-known that high order finite difference WENO schemes have the advantage of simpler coding and smaller computational cost for multi-dimensional problems, compared with finite volume WENO schemes of the same order of accuracy. However a main restriction is that conservative finite difference schemes of third or higher order of accuracy can be achieved *only* on uniform rectangular or smooth curvilinear meshes [14].

Adaptive mesh refinement (AMR) is another useful tool to achieve high accuracy and resolution. AMR automatically refines grid in regions where high resolution is required. The final AMR mesh can contain grids of many refinement levels. Each grid (called *patch*) can be a uniform rectangular or smooth curvilinear mesh. It is easy to apply the high order WENO scheme to each patch. However, interpolation is required for the newly-generated fine mesh and for the boundary conditions at fine-coarse mesh interface. Interpolation in any manner is non-conservative if they are not based on cell averages. Thus, although the numerical schemes in each patch are conservative, the global schemes may not be conservative if the interface between difference patches are not treated conservatively. A conservative interface treatment would require the usage of cell averages. Even for the cell-average values, it is impossible to construct a high order (higher than 2) conservative polynomial in each cell for any refinement ratio.

For numerical simulation of a complex fluid, such as turbulence fluids, both high order discretization and AMR are demanded. It is pointed out in [10] that even ENO schemes as high as sixth-order can be too dissipative for the turbulence if mesh is not fine enough near the shock. Although both WENO and AMR have been proposed for a long time, implementing WENO to the AMR framework is still in its infancy. As far as we have known, we did not find any literatures on this topic. In this paper, we proposed a simple implementation that utilizes the available AMR resource. We compared the high order non-conservative interpolations with the second order conservative interpolations, and found the conservative interpolation were much better.

The outline of the paper is as follows. In Section 2, we described briefly the Berger-Colella's AMR framework for the finite-volume method, and then described our AMR implementation for the finite-difference multi-stage time integration method. In Section 3, we described the high order non-conservative interpolation and second order conservative interpolation that would be used in the AMR framework for WENO schemes. We also pointed out a local flux splitting may be required for conservation of the flux at internal boundaries of AMR.

In Section 4, we presented some numerical examples to test the interpolation schemes for accuracy and conservation.

2 Adaptive mesh refinement

For the sake of completeness, we will first give a brief overview on the Berger-Colella [1] AMR strategy for finite-volume method. Then we describe our AMR implementation for finite-difference method and for multi-stage time integration that is used by WENO scheme. The detail of our AMR implementation was described in [4,5]. To be concise, we here describe only the outline of the implementation.

2.1 Berger-Colella AMR algorithm

Given a grid and solution on it, we first flag those cells that require refinement and then cluster them into several rectangular grids, call *patch*. For each newly-generated patch, we interpolate the solutions from the coarse grid to the new grid. If the old mesh already has some refinement and they share with the new mesh, the solution should be copied from the old fine mesh to the new one. The whole regridding procedure from the old mesh solution to the new mesh solution is called *prolongation*. If the initial conditions are defined analytically, the initial conditions on the new grid are also defined analytically. Each patch can be treated as a single grid. Further refinement can be done recursively until no refinement is needed or the finest refinement level is reached.

During the time integration, the coarse grid is integrated first, and then the fine grid. The boundary conditions of the fine grid are defined by either external boundary conditions, or adjacent sibling patches, or interpolations from the coarse parent grid. After the integration of the fine grid is done, we transfer the more accurate fine grid solutions to the parent coarse grid. This procedure is called *restriction*. The restriction could cause a loss of conservation at the interfaces between a fine and a coarse grid. A flux correction step is proposed in [1] to maintain that the consistent fluxes are used at the fine-coarse interface.

Since the time stepsize is usually determined by the finest grid. Berger and Colella used a local time step method to improve the efficiency. This local time step method allows the fine grid takes smaller time step than the coarse grid before they reach the same time level.

2.2 AMR for finite-difference method

Unlike in the finite-volume method, where the solution variables are always cell-centered (or face-centered for vector field), the solution variables are usually node-centered in a finite-difference method. The node can locate on the domain boundary. We have developed a node-based AMR framework [4] for finite-difference method. The node-based AMR differs from the the cell-centered implementation in two aspects. First, two patches in a level can share the same internal nodes (not only ghost cells) in high dimensions (2D or 3D). A 2-D node-based AMR example is illustrated in Fig. 2.1. Second, a coarse node always shares with a fine node in the same location. This drastically simplify the restriction operation in

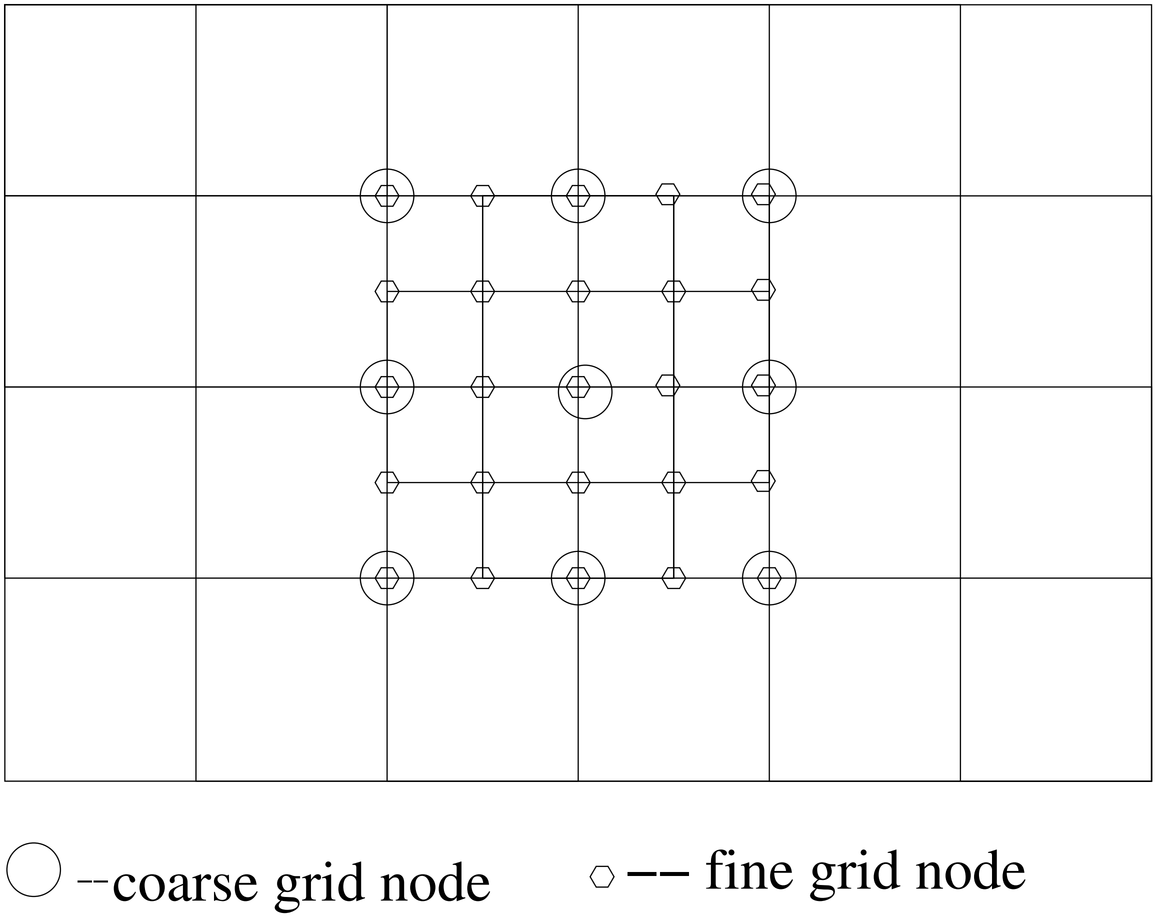

Fig. 2.1. Mesh refinement for finite difference node-based method

AMR framework. The fine grid solution can be copied directly to the coarse grid without any interpolation.

Since the solution variable is node-based, how to flag the cells that need further refinement is different from the cell-centered method. The point-value of the monitor function (or error estimate) is calculated first on each node, and then an average value is evaluated for each cell. If a cell is flagged for further refinement, it is split into fine-cells according to the refinement ratio. For a 2D problem with refinement ratio of 2 in each direction, a cell that contains four nodes can generate nine nodes for a fine grid.

Although relationship between the coarse and fine solutions is simple for the node-based AMR, it is difficult (or impossible?) to make the node-based AMR conservative. First it is difficult to design a conservative interpolation scheme either locally or globally. Second it is difficult to apply the flux correction step proposed by Berger and Colella, because a coarse cell and a fine cell at the fine-coarse interface do not share the same interface.

It seems that the cell-centered AMR framework must be used to make AMR conservative. Compared with the node-centered AMR, one of the disadvantages for the cell-centered AMR is that a coarse grid cell-centered value does not share with a fine grid cell-centered value for the refinement ratio of an even number. The interpolation is required for the restriction operation. Moreover, the solution variables cannot locate at the domain boundary for a cell-centered AMR implementation.

We have implemented both cell-centered and node-centered algorithm in our AMR framework. We found that for if the high order non-conservative interpolation was used, both methods produce almost the same solutions.

2.3 AMR algorithm for multi-stage time integration

The multi-stage time integration is often used to achieve high order temporal accuracy for semi-discretized PDE. The PDE is first transformed into an ODE system by spatial discretization, and then an ODE solver, such as a Runge-kutta or a multistep method, is used to advance in time. This approach, namely the method of lines (MOL) plus an ODE solver, has an advantage of simplicity, both in concept and in coding. The multistage time discretization can achieve arbitrary order of accuracy, depending on the spatial discretization and order of the ODE solver. In this approach, it requires data communications between the AMR grids in each of the intermediate stages. The external boundary conditions also needs to be computed in each stage. Hence we cannot plug this type of solver directly into the Berger-Colella AMR framework.

One issue is how to interpolate the coarse parent grid solutions to get the boundaries data for the current level grid at intermediate stage. There are two approaches to do both time and space interpolation: one is to use only the coarse parent data saved at t_n and t_{n+1}, the other is to use the intermediate state of the coarse parent data. It seems that the second approach would be better than the first one in accuracy. However, we did not see any improvement in our numerical experiment by using the second approach. This may be because the order of the intermediate state is lower than that of the solution at t_{n+1}. Since the second approaches requires more storage, we adopt the first one in our AMR framework. We also improve the first approach that only one space interpolation is required for the whole time step integration. The spatial interpolation is applied to the time derivatives du/dt rather than to the solution u. Therefore, when the interpolation is needed for u at intermediate stage, it can be easily calculated.

With the solutions only at t_n and t_{n+1}, it is impossible to construct an interpolation scheme in time with order higher than two. To overcome this difficulty and test our high order interpolation schemes, we used a locked time step method, described in [9]. The locked time step method disables the time refinement in time integration and allows all of the grids, no matter it is fine or coarse, take the same time steps. It is proved that tested that the locked time step method is only slightly slower than the local time step method. For the locked time step method, the time interpolation is not required and the interpolation order is determined only by the spatial interpolation schemes.

Another issue with the multistage time discretization is how to calculate the flux at the interface, which is required for flux conservation by a correction step on the coarse parent flux. We cannot take the flux at the final stage as the flux for the whole time step. It is also inefficient to do the flux correction for each intermediate stage. We instead obtain the flux for the whole time step by composing the flux at each stage together. The composing is not simply a sum over all of the stages. Take the third order TVD Runge-Kutta solver [15] as an example. The time discretization contains three stages,

$$u^{(1)} = u^n + \Delta t L(u^n, t^n),$$

$$u^{(2)} = \frac{3}{4}u^n + \frac{1}{4}u^{(1)} + \frac{1}{4}\Delta t L(u^{(1)}, t^n + \Delta t),$$
$$u^{n+1} = \frac{1}{3}u^n + \frac{2}{3}u^{(2)} + \frac{2}{3}\Delta t L(u^{(2)}, t^n + \frac{1}{2}\Delta t),$$

where $L(u,t)$ is a spatial discretization operator. As we can see, each stage has a contribution to the total flux of the whole time step. The total flux should be

$$f_{n,n+1} = \frac{1}{6}f(u^n, t^n) + \frac{1}{6}f(u^{(1)}, t^n + \Delta t) + \frac{2}{3}f(u^{(2)}, t^n + \frac{1}{2}\Delta t),$$

where $f(u,t)$ is the flux evaluation at each stage.

3 WENO with AMR

The detail of how to construct a high order finite-difference WENO scheme is described in [6]. As recommended by [17], we used the fifth order WENO scheme combined with the third order Runge-Kutta method in our implementation. We have mentioned that either the high order accuracy or conservation can be lost when high order WENO scheme is applied to adaptive mesh. Sebastian and Shu [14] has proposed a multi-domain finite difference WENO method with high order interpolation at sub-domain interfaces. Their approach achieves the high order accuracy for multi-domain computation but the conservation is lost due to the non-conservative interpolation at the interface.

3.1 High order interpolation

When implementing WENO on an AMR mesh, we can preserve the high order accuracy by using a aomparable high order interpolation method. This high order interpolation must be implemented wherever is required, which includes obtaining the values for the newly-generated fine grid and internal boundary conditions for the integration of the fine patches, and obtaining the values for the coarse parent grid from the fine grid solutions in the restriction step. As we have mentioned in previous section, the time interpolation can be avoided by locked time step method. Hence we focus only on the high order spatial interpolation. The high order method we used is fifth order Lagrange interpolation, which is suggested by [14].

A significant concern with the high order interpolation is that conservative error is introduced. In general, a conservative finite difference scheme for a one dimensional conservation law has the form

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x_j} \left(f_{j+\frac{1}{2}}^n - f_{j-\frac{1}{2}}^n \right). \tag{1}$$

Thus if the scheme is truly conservative and periodic boundary conditions are imposed,

$$\sum_j \Delta x_j u_j^{n+1} = \sum_j \Delta x_j u_j^n, \tag{2}$$

which by induction leads to

$$\sum_j \Delta x_j u_j^n = \sum_j \Delta x_j u_j^0. \tag{3}$$

Therefore, the conservation error is given by

$$CE = \left| \sum_j \Delta x_j u_j^{n+1} - \sum_j \Delta x_j u_j^0 \right|. \tag{4}$$

A similar expression can be defined for the 2D case.

3.2 Conservative interpolation

The conservation is important for nonlinear conservation law that involves discontinuities. It has been proved that if the scheme is conservative and consistent, it is converge to the weak physical solution. Although the conservation error introduced by high order interpolation is small. However this conservation error can accumulate for a long time and affect the accuracy of the numerical solution. In our numerical examples in Section 4, we will see that the non-conservative high order interpolation did produce bad solutions.

The conservation can be achieved if a conservative second order interpolation is used. This interpolation is exactly the same as that which is used for the finite volume method: first a limited slope is computed for each cell-centered coarse value, and then a linear interpolation is applied to calculate the fine grid values within the coarse cell. The conservative restriction operation is to take the average value of the find grid solution within a coarse cell and assign it to the coarse cell. It can be proved that the order of the conservative interpolation is no more than two. This might raise a doubt that the high order accuracy is lost due to the interpolation so that the method is reduced to the low order method. It might be true that a high order method is reduced to a low order method for a fixed multi-domain grid. However, the order of accuracy may not be reduced for an AMR implementation, because the most interested region, which is also the region that contains large error, is always covered by the fine grid between the time steps and no interpolation is needed for those fine grid.

3.3 Flux splitting

The high order WENO schemes are usually implemented via flux splitting method. As suggest by Shu in [6], a global Lax-Friedrichs flux splitting,

$$f^{\pm}(u) = \frac{1}{2}(f(u) \pm \alpha u), \quad \alpha = \max_u |f'(u)|$$

where the maximum is over the whole mesh of the grid, can give as good results as local flux splitting, where the maximum is only over relevant neighborhood cells. We notices that the values of α do affect the values of the final flux at the interface. Different α generate different fluxes. In an AMR grid, two cells in the same refinement level may share the same interface. To be conservative, the flux calculated for each cell must be equal at the interface. However, if each patch calculated their α globally within that patch, the values of α may not be the same, and hence the flux calculated by WENO scheme may not be the same at the same interface. This will cause conservation error.

There are two approach to fix this problem. The first approach is to take a global α over the whole level rather than within a patch. This approach needs communications between different patches in the same level. The other approach is to use the local Lax-Friedrichs flux splitting. Because the α is calculated locally for each cell, the flux calculated at the same interface will be the same. For 5th order WENO finite difference scheme, the α at ith cell is calculated over range from $i-2$ to $i+3$ for the flux at $i+\frac{1}{2}$ interface.

4 Numerical Examples

In this section, we present some examples to test the combination of WENO with AMR.

4.1 Accuracy test

We first solve the linear scalar traveling wave equation,

$$\begin{aligned} u_t + u_x &= 0 \\ u(x,0) &= \sin^m(\pi x), \quad x \in [-1,1], \end{aligned}$$

with periodic boundary conditions.

We test the two-level and three-level refinement with refinement ratio of 3. The norms of error are given by

$$\begin{aligned} L^1 \text{ - error: } ||E||_1 &= \sum_{j=1}^{N} |u(x_j, t^n) - u_j^n| h \\ L^\infty \text{ - error: } ||E||_\infty &= \max_{1\le j\le N} |u(x_j, t^n) - u_j^n|. \end{aligned}$$

where $u(x_j, t^n)$ denotes the exact solution. We stop the integration at $t = 10$.

It is typically difficult for ENO type schemes to achieve full order of accuracy for this problem even for $m = 4$ [16]. To make the problem more suitable for adaptive mesh refinement, we choose $m = 6$, which makes the problem even harder to achieve full-order of accuracy. To reduce the time integration error of the third-order Runge-Kutta method, we used a CFL number of 0.2. Table 1 shows the results obtained for the AMR and uniform grid. It can be seen that WENO with AMR can achieve high order accuracy although only second order interpolation was used. The non-conservative high order interpolation does not improve the numerical solution on an AMR grid.

4.2 Conservation test

We solve the inviscid Burgers' equation with various initial conditions and final computation times:

$$u_t + uu_x = 0, \tag{5}$$

$$u(x,0) = u_0(x). \tag{6}$$

N	Level of refinement	L^1 error	L^1 order	L^∞ error	L^∞ order
80	1	0.443e-2		0.132e-1	
160	1	0.293e-3	3.92	0.544e-3	4.60
240	1	0.439e-4	4.68	0.999e-4	4.18
320	1	0.113e-4	4.72	0.259e-4	4.69
80	2	0.562e-4	3.97	0.128e-3	4.22
80	3	0.742e-5	2.91	0.455e-4	2.58
80	2*	0.604e-4	3.91	0.128e-3	4.22

Table 1

Error estimate of solutions for linear scalar equation $u_t + u_x = 0$ with initial condition $u_0 = \sin^6(\pi x)$ at time $t = 10$. *This is tested with high order non-conservative interpolation, where conservation error is 10^{-5}. The two-level refinement contains about 150 grid points, and three-level refinement contains about 330 grid points. The L_1 and L_∞ order for the refinement grid is calculated against the finest level grid, which may not reflect the accuracy order of the other coarse levels.

The first case is that of a very slow moving shock. The initial condition for this slowly traveling shock is given as

$$u(x,0) = \begin{cases} 1.02, & \text{if } x < 0.03; \\ -1.0, & \text{if } x \geq 0.03. \end{cases}$$

This problem is used in [13] to show that a slowly traveling shock wave has difficulties in passing through the interface between two overlapping subdomains unless the interpolation between the sub-domains is made conservative. We solved it with WENO finite difference scheme combined with two-level refinement.

Figure 4.1-a shows the results of the AMR grid with second order conservative interpolation. Figure 4.1-b shows the results of the AMR grid with high order (5th order) nonconservative interpolation. It is clearly seen that non-conservative high order interpolation did not produce good results. The location of the shock is totally wrong. However, the AMR with conservative interpolation gives almost the same results as the Uniform mesh method with the same resolution. The Uniform grid uses about 42.67 seconds for the total computation while the AMR grid uses only 7.28 seconds.

The second case has been used in [14] to test the ability of the WENO finite difference

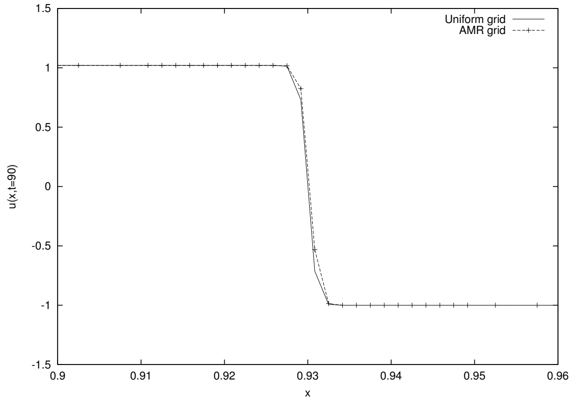

Fig. 4.1-a. WENO on AMR grid with conservative second order interpolation

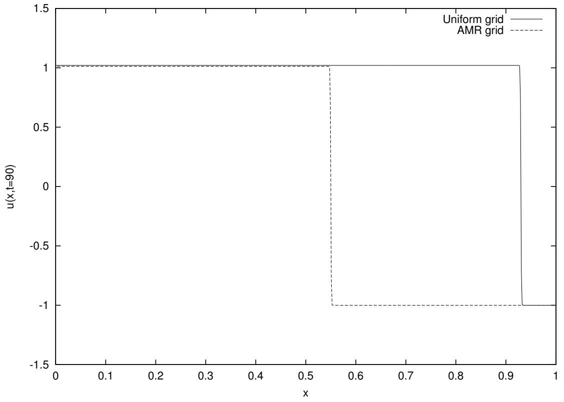

Fig. 4.1-b. WENO on AMR grid with nonconservative fifth order interpolation

method with high order interpolation algorithm for multi-domain. The initial condition is chosen as

$$u(x,0) = 17.4 + 13.3\sin(\pi x), \quad x \in [-1,1].$$

Periodic boundary is used to test the conservation results. In results of [14], the high order interpolation produced noticeable conservation errors, which cause the resulting numerical shock location to lag behind the exact shock location. We tested the high order non-conservative interpolation and second order conservative interpolation in our AMR framework. It shows that the conservative interpolation gives better results although it is only of second order.

Figure 4.3-a shows the results for the conservative interpolation, and Figure 4.3-b shows the results for the non-conservative high order interpolation. It is hard to distinguish between the solutions for the conservative second order interpolation and those generated by the uniform grid with the same resolution, whereas there are noticeable errors for results of high order non-conservative interpolations. Since the problem has periodic boundary condition, we plotted the conservation error for both interpolations in Figure 4.4.

4.3 1-D Euler equations

In this example, we solved Euler equations. For problems that contain only shocks and simple smooth region solution (almost piecewise linear), a good second order non-oscillatory scheme would give satisfactory results. There is little advantage in using higher order methods for such cases. A high order scheme would show its advantage when the solution contains both shocks and complex smooth region structures or for a long time integration. A typical

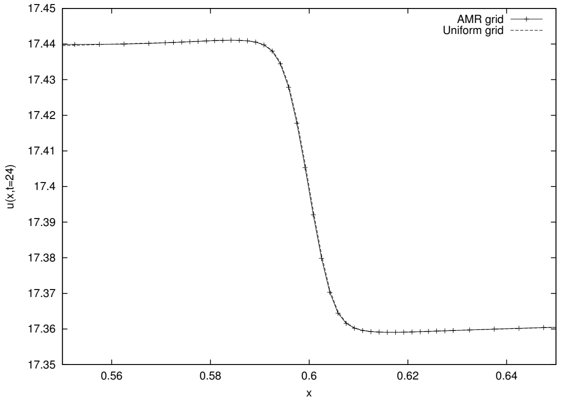

Fig. 4.3-a. WENO on AMR grid with conservative second order interpolation for strong shock problem.

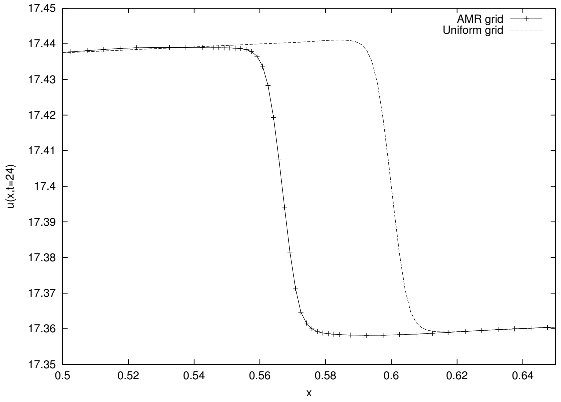

Fig. 4.3-b. WENO on AMR grid with nonconservative fifth order interpolation for strong shock problem.

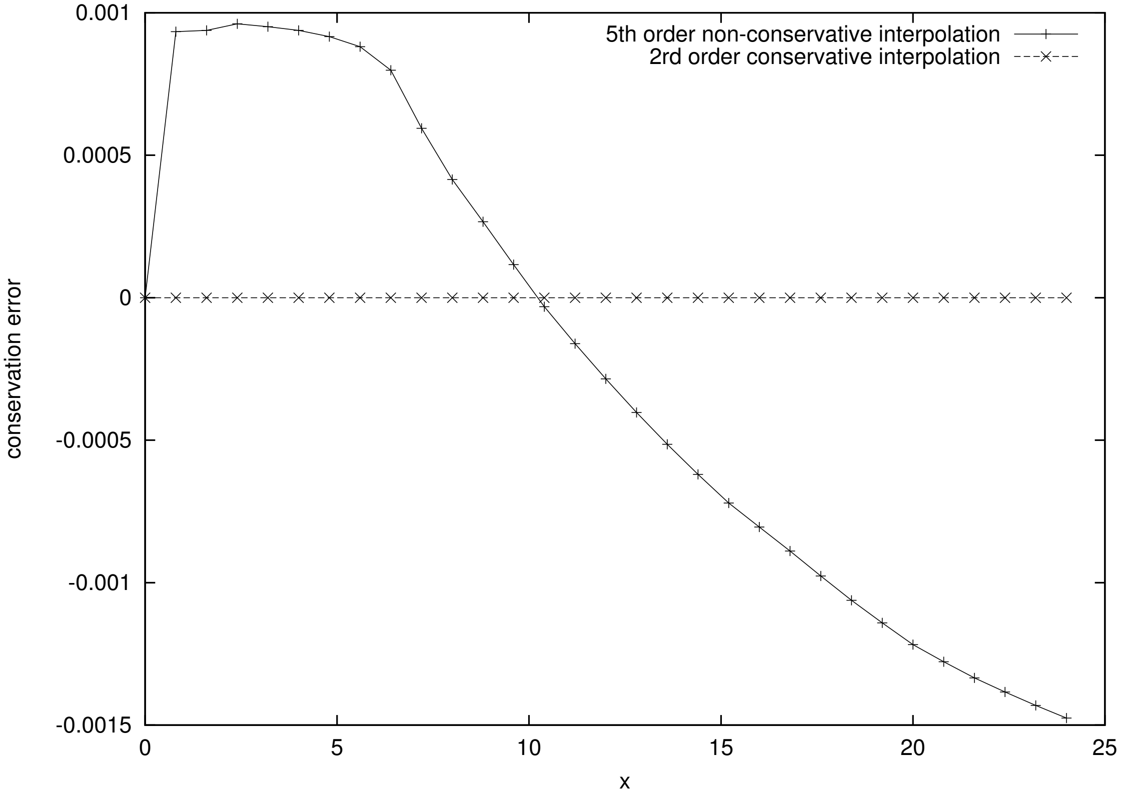

Fig. 4.4. Conservation errors for the strong shock problem

example is given in [15], which is a problem of shock interaction with entropy waves. We solve the Euler equations with a moving Mach 3 shock interacting with sine waves in density, i.e., initially

$$(\rho, v, p) = \begin{cases} (3.857143, 2.629369, 10.3333333), & \text{for } x < -4; \\ (1 + \varepsilon \sin(5x), 0, 1)\,, & \text{for } x \geq -4. \end{cases}$$

where $\varepsilon = 0.2$. The computed density ρ is plotted at $t = 1.8$ against the reference solution, which is calculated by using a uniform mesh with 1800 grid points. We tested the WENO scheme with 4 level refinement. The base grid contains 100 cells and he refinement ratio for each level is 3,3,2 respectively. Therefore, the finest level has equivalent resolution as the uniform grid of 1800 cells. The results are shown in Figure 4.5. We also tested it with a two and three level refinement with the finest level has the same resolution as that of 1800 cells. The results are shown in Figure 4.6. It is hard to notice the difference between different refinement levels.

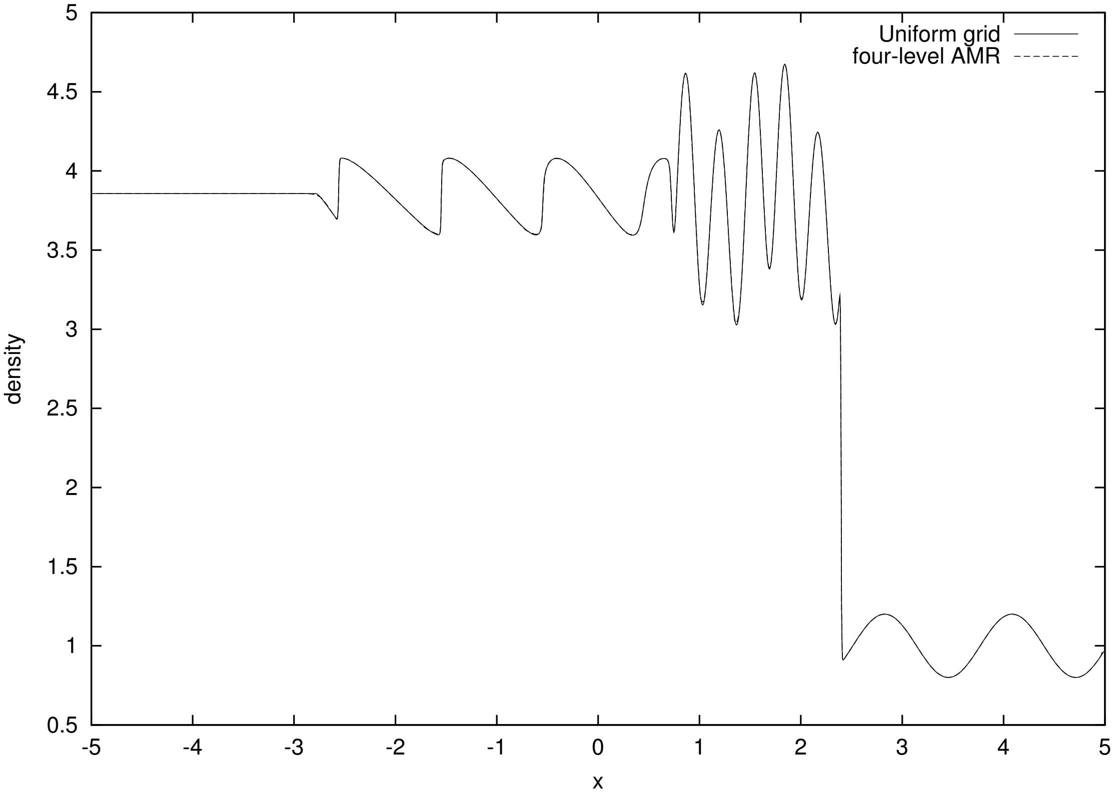

Fig. 4.5. The WENO scheme with four-level refinement for the shock density wave interaction problem. t=1.8.

We also test this problem with high order non-conservative interpolation. The results are shown in Figure 4.7. It is interesting to see that the shock is resolved very well against the uniform grid results, although this method is non-conservative. It might be because the time integration is too short. There are noticeable errors in the complex smooth region, which demonstrated that it is worse than the second order conservative interpolations. We also noticed even larger error if the flux-correction step was implemented for this approach.

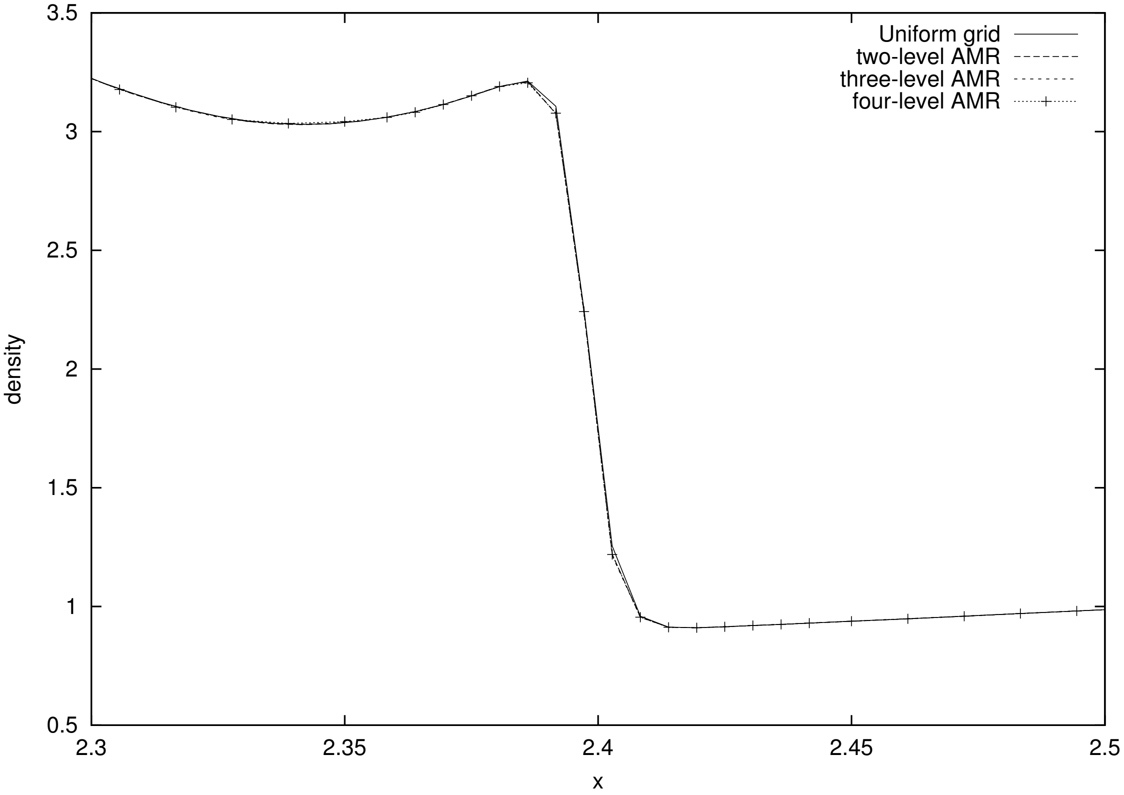

Fig. 4.6. The WENO scheme with AMR of different refinement levels for the shock density wave interaction problem. t=1.8. The CPU time used for the uniform grid is 32.55 seconds, and for AMR grids are 2.15, 3.03, and 5.17 for four-level, three-level and two-level respectively.y

4.4 2-D Euler equations

In this example we solved the 2-D Euler equations. It is the double Mach reflection problem, originally given in [18] and later used often in the literature as a benchmark. The computational domain is [0,4]×[0,1], although typically only the results on [0,3]×[0,1] are shown in the figures. The reflecting wall lies at the bottom, starting from $x = \frac{1}{6}$. The computation is carried out to $t = 0.2$. Figures 4.8-a and 4.8-b show the results of 5th order WENO scheme with three-level refinement. The base grid has spacing 1/60, and the refinement ratio is 3 for each level. Therefore the finest grid has spacing 1/540. The roll-up which represents instability [1] can be observed easily and is comparable to the results of [1] with the finest spacing of 1/800 and second order method.

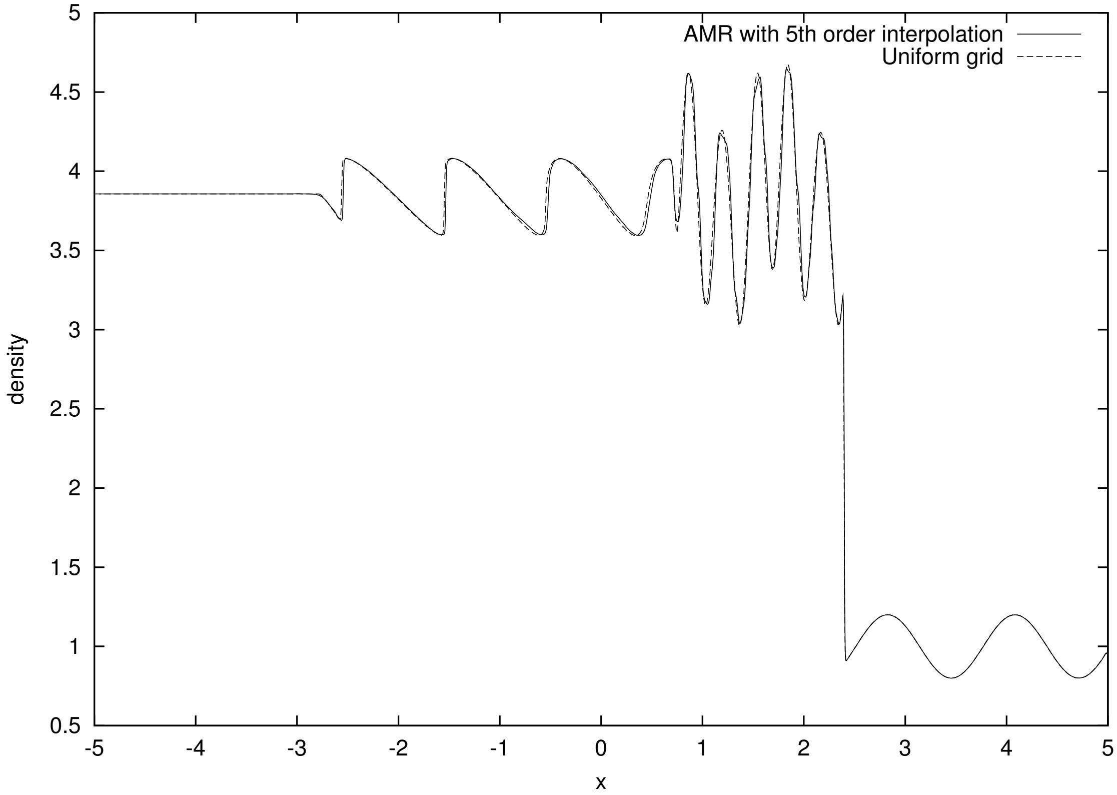

Fig. 4.7. The WENO scheme with four-level refinement for the shock density wave interaction problem. t=1.8.

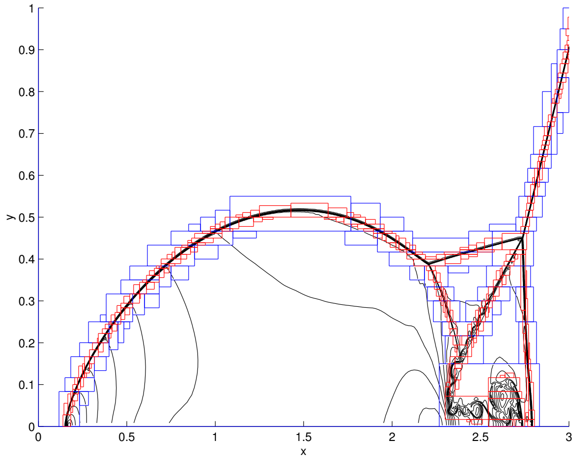

Fig. 4.8-a. Density contours for double Mach reflection problem. 30 contours are used between 1.5 to 21.9.

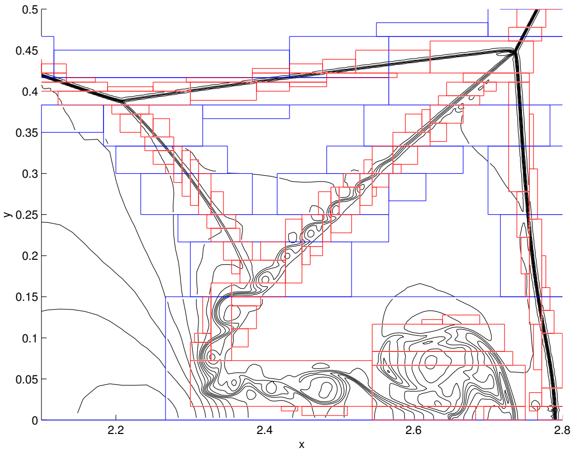

Fig. 4.8-b. A "zoomed in" version of Density contours with refinement.

5 Conclusion

We have implemented high order finite-difference WENO scheme in our AMR framework. Although the finite-difference WENO scheme achieves high order accuracy only for uniform and smooth varying mesh, numerical results have shown that it did achieve high order accuracy even with second order conservative interpolation in AMR framework. We think this may be because the interpolations in AMR occur only in simple smooth region (almost piecewise linear) and second order interpolation there can achieve high accuracy.

We tested the high order non-conservative interpolations for AMR in several numerical examples. The numerical results demonstrated that the high order non-conservative interpolation is no better than the second order conservative interpolation, and in most cases it is even worse.

We strongly recommend using the second order conservative interpolation when implementing WENO to AMR framework.

References

[1] M. J. Berger and P. Colella, Local adaptive mesh refinement for shock hydrodynamics, *J. Comput. Phys.* 82 (1989), 64-84.

[2] M. J. Berger and J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, *J. Comput. Phys.* 53 (1984), 484-512.

[3] M. J. Berger and I. Rigoutsos, An algorithm for point clustering and grid generation, *IEEE Trans. on Systems, Man, and Cybernetics*, 21 (1991).

[4] Shengtai Li and Mac Hyman, Solution adapted nested grid refinement for 2-D PDEs, 1998, Technical Report, LA-UR-98-5463, Los Alamos National Lab.

[5] Shengtai Li and Mac Hyman, Interactive and dynamic control of adaptive mesh refinement with nested hierarchical grids, 1998, Technical Report, LA-UR-98-5462, Los Alamos National Lab.

[6] G. S. Jiang and C. W. Shu, efficient implementation of weighted ENO schemes, *J. Comt. Phys.* 126 (1996), 202.

[7] A. Kurganov and D. Levy, A third order semi-discrete central scheme for conservation laws and convection-diffusion equations, *SIAM J. Sci. Comput.*, 22 (2000), 1461-1488.

[8] D. Levy, G. Puppo and G. Russo, A third order central WENO scheme for 2D conservation laws, *Appl. Numer. Math.* 33 (2000), 407-414.

[9] S. Li and J. M. Hyman, Local time step versus locked time step, Technical report, 2002, Los Alamos National Lab.

[10] P. Moin and K. Mahesh, Direct numerical simulation: a tool in turbulence research, *Annu. Rev. Fluid Mach.* 30 (1998), 539-578.

[11] X.-D. Liu, S. Osher and T. Chan, Weighted essentially non-oscillatory schemes, *J. Comput. Phys.*, 115 (1994), 200-212

[12] D. Balsara and C. W. Shu, Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy, *J. Comput. Phys.*, 160 (2000), 405-452.

[13] E. Part-Enander and B. Sjogreen, Conservative and non-conservative interpolation between overlapping grids for finite volume solutions of hyperbolic problems, *Computer fluids*, 23 (1994), 551-574.

[14] K. Sebastian and C. W, Shu, Multi domain WENO finite difference method with interpolation at sub-domain interfaces. Journal of Scientific Computing, to appear 2003.

[15] C. W. Shu and S. J. Osher, Efficient implementation of essentially non-oscillatory shock capturing schemes II, *J. Comput. Phys.*, 83(1989), 32-78.

[16] C. W. Shu, Numerical experiments on the accuracy of ENO and modified ENO schemes, *J. Sci. Comput.*, 5 (1990), 127-149.

[17] C. W. Shu, High order finite difference and finite volume WENO schemes and discontinuous Galerkin method for CFD, International Journal of Computational Fluid Dynamics, to appear.

[18] P. R. Woodward and P. Colella, The numerical simulation of two-dimensional fluid with strong shocks, *J. Comput. Phys.*54 (1984), 115-173.