Parallel adaptive mesh refinement method based on WENO finite difference scheme for the simulation of multi-dimensional detonation[☆]

Cheng Wang[a], XinZhuang Dong[a], Chi-Wang Shu[b,*]

[a]State Key Laboratory of Explosion Science and Technology, Beijing Institute of Technology, Beijing 100081, PR China
[b]Division of Applied Mathematics, Brown University, Providence, RI 02912, USA

Abstract

For numerical simulation of detonation, computational cost using uniform meshes is large due to the vast separation in both time and space scales. Adaptive mesh refinement (AMR) is advantageous for problems with vastly different scales. This paper aims to propose an AMR method with high order accuracy for numerical investigation of multi-dimensional detonation. A well-designed AMR method based on finite difference weighted essentially non-oscillatory (WENO) scheme, named as AMR&WENO is proposed. A new cell-based data structure is used to organize the adaptive meshes. The new data structure makes it possible for cells to communicate with each other quickly and easily. In order to develop an AMR method with high order accuracy, high order prolongations in both space and time are utilized in the data prolongation procedure. Based on the message passing interface (MPI) platform, we have developed a workload balancing parallel AMR&WENO code using the Hilbert space-filling curve algorithm. Our numerical experiments with detonation simulations indicate that the AMR&WENO is accurate and have a high resolution. Moreover, we evaluate and compare the performance between the uniform mesh WENO scheme and the parallel AMR&WENO

method. The comparison results provide us further insight into the high performance of the parallel AMR&WENO method.

1. Introduction

Detonation involves complex interactions between reactive chemical kinetics and fluid dynamics. It consists of a precursor shock wave that propagates into the unreacted medium at supersonic speed with a thin reaction zone immediately behind the shock. The research on detonation phenomena has lasted for more than one hundred years, yet even today many phenomena are still not very well understood. Because of the limited insight that experimental observations can provide, numerical simulations play a more and more important role in the investigation of detonation phenomena.

Detonation waves exhibit non-negligible multi-dimensional sub-structures. These sub-structures are in the millimeter range which is much smaller than the size of the physical geometry. The additional stiff source terms modeling chemical reactions introduce extremely small time scales into the flow field which are much smaller than the fluid-dynamical time scale. There is a requirement of a locally refined resolution for the numerical method, due to the separation in both time and space scales. The adaptive mesh refinement (AMR) method has an advantage for solving such problems with separation scales. It automatically refines the mesh locally where a fine resolution is needed. The AMR method was first developed by Berger et al. [1-2] to solve hyperbolic partial differential equations. Nowadays it is widely used in detonation simulation and has been validated to be an effective tool [3-11].

Recently, weighted essentially non-oscillatory (WENO) schemes have attracted increasing attention in the investigation of detonation owing to their advantage of high order accuracy, high resolution and essentially non-oscillatory shock resolution [12-14]. As is well known, WENO schemes can resolve complex solution structures with a relatively coarse numerical mesh. However, the mesh should still be refined enough to resolve detonation phenomena. High order conservative finite difference WENO schemes are very efficient in multi-dimensions in comparison with finite volume WENO schemes, yet they require uniform or smooth varying meshes. It is expensive in the investigation of detonation waves which would only need a locally refined resolution.

2

In order to develop an efficient method for detonation simulation, this paper aims to combine AMR methods with WENO schemes. The coupling of AMR methods and WENO schemes has been extensively studied. Li et al. [15] used a fifth order Lagrange interpolation for the prolongation in the space and a locked time step method in time. The locked time step method avoids time refinement, with the time step size limited by the CFL condition of the finest mesh. It may result in a waste of the computational effort. Baeza et al. [16] adopted the linear interpolation in time which would reduce the temporal accuracy of the scheme. Shen et al. [17] proposed an AMR method based on high order WENO schemes for hyperbolic conservation laws. High order prolongation is used in both space (WENO interpolation) and time (Hermite interpolation). The AMR method with high order prolongation is proved to be an accurate and robust method. However, due to the adoption of block-based data structure, the method in [17] may involve unprofitable refinement and is inflexible in dealing with complicated geometry. Also, the simulation is performed on hyperbolic conservation laws without considering possible complications from stiff source terms.

This paper focuses on developing an AMR method base on the fifth order WENO finite difference scheme for detonation problems, which is a generalization of the method in [17]. The outline of the paper is as follows. In Section 2, an appropriate model for gas detonation is proposed. The one-step Arrhenius chemistry model is utilized in this paper. Section 3 gives a description of the WENO scheme and introduces the detailed implementation of the parallel AMR&WENO method. In Section 4, the parallel AMR&WENO method is used to investigate a series of multi-dimensional detonation problems and a detailed analysis of the numerical results is provided.

2. Governing equations

The reaction mechanism adopted in this paper is the one-step reaction model. It is well known that one-step reaction models cannot reveal the detailed reaction chemistry of detonations and cannot give quantitatively correct description of detonation. However, it does have the practical advantage in terms of computational cost. In the investigation of detonation, detailed reaction models consume huge amount of computer resource. It is not realistic to use the detailed reaction model to simulate detonation in large domains or for high dimensional problems. In many situations, one-step reaction models are good choices as a compromise between cost and

performance.

This paper aims to develop and test a numerical method, hence the simple one-step reaction model is a good testing model. This model for detonation propagation in premixed gases neglects the effect of viscosity, the heat transfer, the diffusion and body forces. It can be written as the Euler equations with reactive source terms. In three-dimensional Cartesian coordinates, these equations are given as follows

$$
\begin{aligned}
&\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial H}{\partial z} = S \\
&U = (\rho, \rho u, \rho v, \rho w, \rho E, \rho Y)^T \\
&F = (\rho u, \rho u^2 + p, \rho uv, \rho uw, (\rho E + p)u, \rho uY)^T \\
&G = (\rho v, \rho uv, \rho v^2 + p, \rho vw, (\rho E + p)v, \rho vY)^T \\
&H = (\rho w, \rho uw, \rho vw, \rho w^2 + p, (\rho E + p)w, \rho wY)^T \\
&S = (0, 0, 0, 0, 0, \omega)^T
\end{aligned}
\tag{1}
$$

where u, v and w are the components of the fluid velocity in the x, y and z directions, respectively. ρ is the density, p is the pressure, E is the total energy per unit volume, and Y is the mass fraction of the reactant.

The total energy E is defined as following

$$
E = \frac{p}{\rho(\gamma - 1)} + \frac{(u^2 + v^2 + w^2)}{2} + qY
\tag{2}
$$

where q is the heat production in the reaction, and γ is the ratio of specific heat.

The source term S is assumed to have an Arrhenius form and is written as follows

$$
\omega = -K\rho Y \exp\left(-\frac{E_a}{RT}\right)
\tag{3}
$$

where E_a is the activation energy and K is the constant rate coefficient. For perfect gas, the equation of state is $p = \rho RT$.

3. The numerical method

As mentioned above, the AMR method has significant advantages for detonation problems owing to its local high resolution. The WENO scheme

is outstanding in the numerical simulation of detonation because of its high order accuracy and essentially non-oscillatory shock resolution. It is very natural to combine them to study detonation problems. Nowadays, both AMR and WENO methods have attracted increasing attention in the numerical simulation of detonation. Although there are many investigations about the combination of them, few of these studies focus on solving detonation problems, especial in three dimensions. In this paper, an AMR method based on finite difference WENO schemes is proposed to solve multi-dimensional detonation problems.

3.1. Review of finite difference WENO schemes

In this section, we briefly describe the framework of WENO schemes for solving hyperbolic conservation laws. More details of the high order finite difference WENO schemes can be found in the literature [18-19]. Consider a uniform mesh with grid points $x_i = i\Delta x$, $i = 0, \ldots, nx$, $y_j = j\Delta y$, $j = 0, \ldots, ny$ and $z_k = k\Delta z$, $k = 0, \ldots, nz$. The cell boundaries are given by $x_{i+1/2} = x_i + \Delta x/2$, $y_{i+1/2} = y_i + \Delta y/2$, $z_{i+1/2} = z_i + \Delta z/2$. The partial differential equation at the grid points is written as the following form.

$$\frac{dU(x_i, y_j, z_k, t)}{dt} = \left(-\frac{\partial F(U)}{\partial x} - \frac{\partial G(U)}{\partial y} - \frac{\partial H(U)}{\partial z} + S \right) |_{x=x_i, y=y_j, z=z_k} \quad (4)$$

We use a finite difference scheme to approximate the Eq.4.

$$\left(\frac{\partial U}{\partial t} \right)_{i,j,k} = -\frac{\widehat{F}_{i+1/2,j,k} - \widehat{F}_{i-1/2,j,k}}{\Delta x} - \frac{\widehat{G}_{i,j+1/2,k} - \widehat{G}_{i,j-1/2,k}}{\Delta y}$$
$$- \frac{\widehat{H}_{i,j,k+1/2} - \widehat{H}_{i,j,k-1/2}}{\Delta z} + S_{i,j,k} \quad (5)$$

where U is an approximation to the point value $U(x_i, y_j, z_k, t)$. $\widehat{F}_{i\pm1/2,j,k}$, $\widehat{G}_{i,j\pm1/2,k}$ and $\widehat{H}_{i,j,k\pm1/2}$ are numerical fluxes on the cell boundary and can be obtained from WENO reconstruction.

Take the numerical flux $\widehat{F}_{i\pm1/2,j,k}$ and fifth order accuracy as an example, the WENO reconstruction is implemented as follows. The fifth order WENO scheme uses 5-point stencil to approximate the numerical flux, which is divided into three 3-point sub-stencils S_0, S_1 and S_2.

First, the Lax-Friedrichs flux splitting is performed.

$$F^{\pm} = \frac{1}{2}\left(F(U) \pm \alpha U\right) \tag{6}$$

where α is taken as $\alpha = \max_U |F'(U)|$.

Then two numerical fluxes $\widehat{F}^+_{i+1/2,j,k}$ and $\widehat{F}^-_{i+1/2,j,k}$ are constructed to approximate F^+ and F^- whose sum is the numerical flux $\widehat{F}_{i+1/2,j,k}$. As the construction procedure of $\widehat{F}^+_{i+1/2,j,k}$ and $\widehat{F}^-_{i+1/2,j,k}$ is mirror symmetric with respect to $i + 1/2$, we only give the details of the construction of $\widehat{F}^+_{i+1/2,j,k}$ in the following.

$$\widehat{F}^+_{i+1/2,j,k} = \sum_{r=0}^{2} \omega_r \widehat{F}^{(r)}_{i+1/2,j,k} \tag{7}$$

where ω_r are the nonlinear weights satisfying $\omega_0 + \omega_1 + \omega_2 = 1$ and are defined in the following form

$$\omega_r = \frac{\alpha_r}{\sum_{s=0}^{2} \alpha_s}, \quad \alpha_r = \frac{d_r}{(\epsilon + \beta_r)^2} \tag{8}$$

where d_r are the linear weights and β_r are the smoothness indicators. ϵ is a small constant used to avoid the denominator to become zero. The fluxes for the three sub-stencils have the form

$$
\begin{aligned}
\widehat{F}^{(0)}_{i+1/2,j,k} &= \frac{1}{3}F_{i-2,j,k} - \frac{7}{6}F_{i-1,j,k} + \frac{11}{6}F_{i,j,k} \\
\widehat{F}^{(1)}_{i+1/2,j,k} &= -\frac{1}{6}F_{i-1,j,k} - \frac{5}{6}F_{i,j,k} + \frac{1}{3}F_{i+1,j,k} \\
\widehat{F}^{(2)}_{i+1/2,j,k} &= \frac{1}{3}F_{i,j,k} + \frac{5}{6}F_{i+1,j,k} - \frac{1}{6}F_{i+2,j,k}
\end{aligned} \tag{9}
$$

The linear weights are $d_0 = 3/10, d_1 = 3/5, d_2 = 1/10$, and the smoothness indicators are given as follows

$$
\begin{aligned}
\beta_0 &= \frac{13}{12}(F_{i-2,j,k} - 2F_{i-1,j,k} + F_{i,j,k}) + \frac{1}{4}(F_{i-2,j,k} - 4F_{i-1,j,k} + 3F_{i,j,k}) \\
\beta_1 &= \frac{13}{12}(F_{i-1,j,k} - 2F_{i,j,k} + F_{i+1,j,k}) + \frac{1}{4}(F_{i-1,j,k} - F_{i+1,j,k}) \\
\beta_2 &= \frac{13}{12}(F_{i,j,k} - 2F_{i+1,j,k} + F_{i+2,j,k}) + \frac{1}{4}(3F_{i,j,k} - 4F_{i+1,j,k} + F_{i+2,j,k})
\end{aligned} \tag{10}
$$

The conservative updating of U needs the numerical fluxes $\widehat{F}^+_{i+1/2,j,k}$ and $\widehat{F}^+_{i-1/2,j,k}$, involving 6 points in the x-direction around the point (i,j,k). In three dimensions, the numerical fluxes $\widehat{G}^+_{i,j\pm1/2,k}$ and $\widehat{H}^+_{i,j,k\pm1/2}$ in the y and z directions can be obtained by utilizing the same procedure. The calculation of $\left(\frac{\partial U}{\partial t}\right)_{i,j,k}$ involves 16 points in all three directions. Considering the calculations on two adjacent points (i,j,k) and $(i-1,j,k)$, it can be observed that they share many sub-stencils. If we implement the WENO procedure on each point independently (cell-by-cell type), it would result in the repeated calculation on the shared sub-stencils. It appears to be more efficient to implement the WENO procedure for points in the same dimension jointly (dimension by dimension type).

If we denote the right hand side of Eq.5 as $L(U_{i,j,k})$ and use the third-order TVD Runge-Kutta scheme for temporal discretization, then the fully discrete scheme can be written in the following form

$$
\begin{aligned}
U^{(1)}_{i,j,k} &= U^n_{i,j,k} + \Delta t L(U^n_{i,j,k}) \\
U^{(2)}_{i,j,k} &= \frac{3}{4}U^n_{i,j,k} + \frac{1}{4}U^{(1)}_{i,j,k} + \frac{1}{4}\Delta t L(U^{(1)}_{i,j,k}) \\
U^{n+1}_{i,j,k} &= \frac{1}{3}U^n_{i,j,k} + \frac{2}{3}U^{(2)}_{i,j,k} + \frac{2}{3}\Delta t L(U^{(2)}_{i,j,k})
\end{aligned}
\tag{11}
$$

The reconstruction for the fluxes involves information on a wide stencil with a number of nearby grid points and this makes the AMR implementation more complicated. A proper organization of data is necessary. Local time refinement and the Runge-Kutta sub-steps also make the AMR implementation more complicated. The details of the implementation will be described in the following subsections.

3.2. AMR data structure

Different data structures used to track mesh connectivity make a big difference in the performance of the AMR method. The data structure for AMR can be generally classified into two categories, block-based structure and cell-based structure. Fig. 1 shows the sketch of both the block-based structure and the cell-based structure. Fig. 1(a) demonstrates uniform cells on level 0 covering the overall computational domain. Cells with black dots require higher resolution and need to be refined. Fig. 1(b) and Fig. 1(c) depict the refinement based on block-based structure and cell-based structure, respectively. As shown in Fig. 1(b), a sequence of structured blocks is

7

nested on different levels patched onto each other. The block represents a rectangular logical domain containing many cells. On the same refinement level, the meshes of the block are refined by the same refinement ratio in both time and space. Refinement based on cell-based structure is shown in Fig. 1(c). Each cell is refined individually. Both cell-based structure and block-based structure can be abstract to the logical hierarchical tree structure as shown in Fig. 1(d). The nodes of the tree represent the blocks for block-based structure and individual cells for cell-based structure. There is some advantage for block-based structure. Because the tree structure resulted from block-based structure is lighter than that of cell-based structure, the computational effort to access the node and the memory overhead to maintain the tree structure information of block-based structure are smaller than that of cell-based structure. The original solver can be directly used on the block without any modification. However, block-based structure relies on sophisticated clustering algorithm to organize individual cells into a block. The clustering algorithm makes cells that do not need to be refined also involved in the block and this results in a waste of computational resource. On the same refinement level, the blocks bond with each other such as in Fig. 1(e). Blocks communicate with each other in order to exchange boundary information in the process of calculation. As the procedure of WENO scheme involves many points and needs a lot of boundary information, the information exchange between blocks is very large. Cell-based structure avoids unnecessary refinement and it readily achieves the local refinement without clustering algorithm. It is difficult to say which approach is better in practice, but the cell-based structure has high flexibility and efficiency in theory. We adopt the cell-based structure in this work.

As mentioned above, the tree structure is heavy for the cell-based structure. Different data storage structures have distinct performance on computational effort and the memory overhead. Traditional tree data structure used to store the cell data has a high computational effort and high memory overhead. A data structure with relatively low computational effort is proposed by Khokhlov et al. [20] named as the fully threaded tree (FTT) structure. FTT adds pointers in order to achieve the connection among cells. Pointers stored in the cell include a pointer for its parent cell, pointers for its child cells and pointers for neighboring cells. For the FTT structure, it is possible to access the neighbors of a cell without a search operation owing to the pointers, but the memory cost is large. As the WENO scheme involves a lot of neighboring cells, the memory cost used to store pointers is
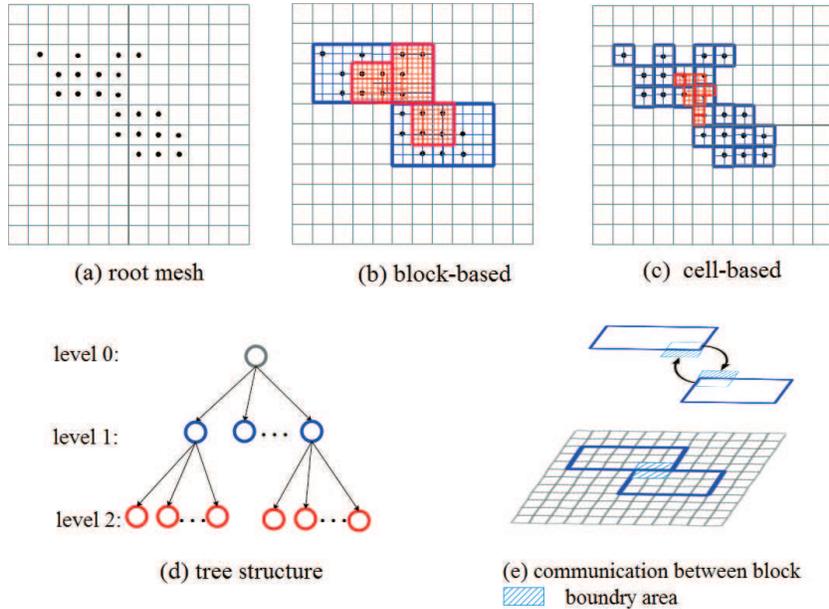
Fig. 1: Sketch of block-based structure and cell-based structure.

especially large. In recent research [5], a new cell-based structured adaptive mesh refinement (CSAMR) data structure is proposed aimed to decrease the memory cost. It uses Cartesian-like indices to identify each cell. A simple hash table structure is used to maintain the cell data. The memory overhead is significantly reduced. At the same time, the information on the parent, child and neighboring cells can be accessed with low computational effort. However, this approach also has its own disadvantages. The computational effort to create a hash table may be tremendous if the reserved memory for the hash table is too small. The computational effort and memory overhead heavily rely on the size of the hash table. It is difficult to obtain a reasonable hash table size to balance the computational effort and memory overhead.

As mentioned in sub-section 3.1, the WENO scheme involves a lot of neighboring cells and needs a dimension by dimension type treatment. The feature of the WENO scheme requires the data structure to have high performance on accessing neighboring cells and to be easy in implementing the dimension-by-dimension type calculation. In this paper, we propose a new AMR data structure based on the WENO scheme. This data structure is similar to CSAMR except for the hash table. The details of the data structure

9

are outlined in following.

Take two dimensions for an example, the refinement ratio is $r = 3$. A square spatial domain with dimension $L_x \times L_y$ is firstly discretized into a coarse, uniform computational mesh (the level of refinement is 0) called "root cells". The cell on the finer mesh is called a child cell. It is created to focus computational efforts on parts of the domain. The cells on the domain are shown in Fig. 2(a). The center of the cell with dimension $(\Delta x, \Delta y)$ is located at the grid node (x_c, y_c). Standard structured cell indexing is (i, j), with $i = 0, \ldots, nx$ and $j = 0, \ldots, ny$.
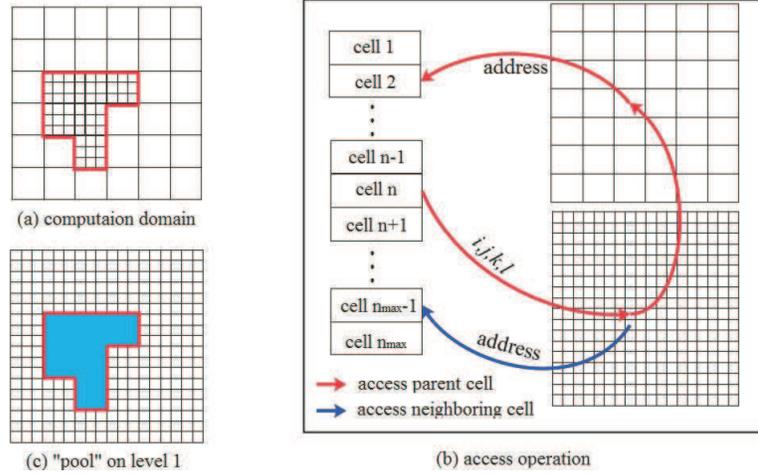
Fig. 2: Sketch of the proposed cell-based structure.

We use Cartesian-like indices and the refinement level l to identify each cell on the l-level which is similar to CSAMR. According to the definition above, the centroid $(x_c, y_c)_l$ and dimension $(\Delta x, \Delta y)_l$ of the an l-level cell with an index $(i, j)_l$ is given as follows

$$(x_c, y_c)_l = ((i + \frac{1}{3}) \times \frac{L_x}{3^l}, (j + \frac{1}{3}) \times \frac{L_y}{3^l})$$
$$(\Delta x, \Delta y)_l = (\frac{L_x}{3^l}, \frac{L_y}{3^l}) \tag{12}$$

The indices of the child cell and the parent cell then can be calculated from

$$(i, j)_{l-1} = (3 \times i + n, 3 \times j + m)$$
$$(i, j)_{l+1} = ([\frac{i}{3}], [\frac{j}{3}]) \tag{13}$$

10

where, $[a]$ denotes the integer value of the quantity a.

Data storage in each cell is outlined as following

(1) The indices of the cell: (i, j);

(2) Cell level: l;

(3) Cell flag: $flag$ (if the cell is not refined, then set $flag = 0$, else $flag = 1$);

(4) The numerical solution: U;

(5) The time derivative of the solution: $D(U)_t$;

On every refinement level, a Cartesian-like list is created to store the address information of cells. We define the list as $C_l(i, j) =$ cell-address. We can use the lists and the indices to access arbitrary cells without any access operation. For example, neighboring cells in the y-direction can be accessed directly by the list $C_l(i, j - 1)$ and the parent cell can be accessed by $C_{l-1}([\frac{i}{3}], [\frac{j}{3}])$ as shown in Fig. 2(b).

In the process of calculation, the Cartesian-like list is used to cluster the cells on the same level l into an irregular domain referred as "pool" as shown in Fig. 2(c). The cells in the "pool" have a same size. The "pool" can be considered as an irregular computational domain and this makes it possible to implement the WENO procedure in the dimension by dimension style. This data structure has a higher memory overhead compared with CSAMR, but it has higher efficiency when used in the WENO scheme.

3.3. Criterion and prolongations

In the implementation of AMR, the first issue is to determine where an adaptive refinement mesh is needed. A refinement criterion is used to address this issue. There are many approaches to construct the refinement criterion. A popular criterion is the estimation of the local truncation error based on Richardson extrapolation [1-2]. However, this is a computationally expensive procedure. A physics-based criterion is often used in the simulation of detonation. Ji et al. [5] used cell-size-weighted velocity divergence and curl as the refinement criterion. The criterion used in [6] is based on local gradients of fluid density and temperature. Wilkening et al. [7] used the difference of velocity between neighboring nodes. In this paper, the gradients of both the mass fraction Y and the pressure p are employed as the criterion. The cells are refined if $|grad(Y_{i,j,k})| \geq \alpha_{cr}$ or $|grad(p_{i,j,k})| \geq \beta_{cr}$, where α_{cr} and β_{cr} are products of the average gradients of all cells in the computational domain and a constant coefficient whose value should be fixed after experiments. A buffer zone is created to ensure shocks can not propagate out of the refinement regions in one time step. The cells in the refinement regions and the

11

cells in the buffer zone are flagged with $flag = 1$. The neighbors of these cells defined as "ghost cells" are the internal boundary. Their flags are set to $flag = 2$. These cells in the refinement areas and buffer areas are refined in space. The time step is also refined with the same refinement ratio.

During the refinement, a new cell C_l as the child of the cell C_{l-1} is created. The data in the child cell C_l can be obtained in two ways. If the child cell is covered by a previously existing cell, the data would be copied from the existing cell directly. Otherwise, the data is prolonged from the parent cell. In this paper, the scheme used to discretize in space is of fifth order accuracy. Because a low order prolongation may reduce the accuracy of AMR, the fifth order WENO interpolation prolongation in [21] is used to prolong the data from C_{l-1} to C_l. For multi-dimensional problems, the implementation of the WENO interpolation utilizes a "dimension by dimension" style. The prolongation in three dimensions can be written as following

$$U_l(i, j, k) = WI(U_{l-1}([\frac{i}{3}] + m, [\frac{j}{3}] + n, [\frac{k}{3}] + p))$$
$$with, m = -3, \ldots, -3, n = -3, \ldots, -3, p = -3, \ldots, -3$$

(14)

where WI denotes the WENO interpolation operator.

3.4. Internal boundary

The refinement time step is defined as in Eq.(15) below. It is based on the enforcement of a global CFL condition. The time step on the refinement level l is refined by the refinement ratio r

$$\Delta t_l = \frac{\Delta t}{r^l}$$

(15)

where Δt is the time step of the "root cells".

Because of the refinement of the time step, the boundary condition on every intermediate time stage should be supplied before the cells are integrated on the fine mesh. Since the internal boundary of the refinement area does not coincide with the physical boundary, the internal boundary condition should be obtained by constructing the data on "ghost cells" at every intermediate time stage.

One issue arising is how to construct the data on ghost cells at an intermediate stage. In addressing this issue, two main approaches are considered. One is a uniform time step on all refinement levels, used in [5]. It avoids the

construction on "ghost cells" at an intermediate stage. But this approach results in a waste of computational effort, because the time step must be small enough to satisfy the CFL restriction of the finest mesh. Another approach is to construct the data using the parent data at two time stages. The second approach is adopted in this paper. The Hermite interpolation method is used to construct the data of "ghost cells". We give a brief description of the method. More details can be found in [17].

We take two time stages t_n and t_{n+1} with a time step Δt as an example. The data of parent cells at the two time stages are known. They are written as U_l^n, U_l^{n+1}, $D_t(U_l)^n$ and $D_t(U_l)^{n+1}$. The prolongation in space mentioned above is used to construct the data of the "ghost cells" at these two time stages. Utilizing the data U_g^n, U_g^{n+1}, $D_t(U_g)^n$ and $D_t(U_g)^{n+1}$ of the "ghost cells", a cubic polynomial $M(t)$ is constructed by Hermite interpolation

$$
\begin{aligned}
M(t^n) = U_g^n, \quad M(t^{n+1}) = U_g^{n+1} \\
M'(t^n) = D_t(U_g)^n, \quad M'(t^{n+1}) = D_t(U_g)^{n+1}
\end{aligned}
\tag{16}
$$

For an arbitrary time stage $t \in [t^n, t^{n+1}]$, an approximation of the ghost cell data is expressed as following

$$
U_g(t) \approx M(t)
\tag{17}
$$

The data of the TVD-RK sub-steps are also constructed

$$
\begin{aligned}
U_g(t) &= M(t) \\
U_g^{(1)}(t) &= M(t) + \Delta t_l M'(t) \\
U_g^{(2)}(t) &= M(t) + \frac{\Delta t_l}{2} M'(t) + \frac{\Delta t_l^2}{4} M''(t)
\end{aligned}
\tag{18}
$$

When the internal boundary condition is supplied, the child cells on the level $l + 1$ are integrated with a refinement time step Δt_l from t^n to t^{n+1}. After the integration on the child cells, the data of the child cells are used to update the data on parents cells.

When data on the child cells iterate to t^{n+1}, a cell to cell replacement process is implemented to update the parents cells. For example, the data of the parent cell with centroid located at (x_c, y_c, z_c) are replaced directly by that of the child cell with the same centroid.

3.5. Parallelization strategy

One of the major goals of this paper is to design a parallel AMR&WENO method based on the MPI platform to perform well on multiprocessor machines. It is well known that AMR refines the mesh locally. As the distribution of refinement cells is nonuniform and changes over time, the computational efforts dynamically change over the global domain. Static partition of the domain would result in a workload imbalance. Various kinds of dynamic partition approach are proposed to balance the workload for AMR computations such as the space filling curve algorithm [22], the bin packing algorithm [23] and so on. The space-filling curve approach is adopted in this paper. The details of the implementation are outlined in the following.

(a) mapping from cells to curve   (b) partition of domain
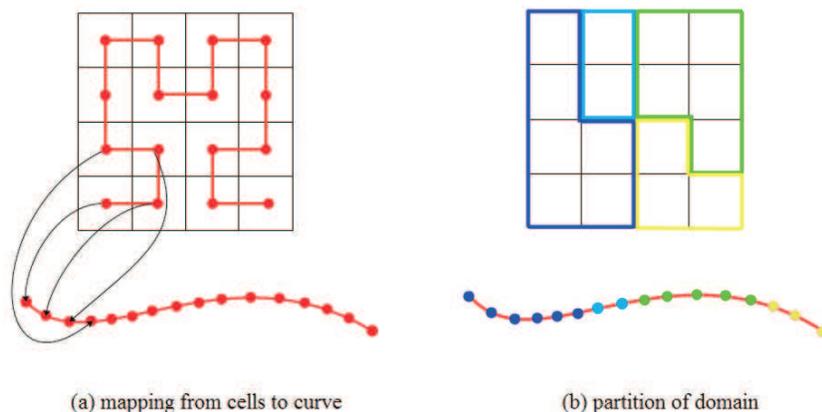
Fig. 3: Sketch of the Hilbert space-filling curve algorithm.

The computational domain is initially divided into cells at level 0 referred as "root cells". We assign root cells and their child cells on every refinement level to the same processor and count the workload of the child cells in the corresponding root cells. The domain partition problem is now simplified to the assignment problem of root cells with different workload to different processors. The Hilbert space-filling curve load balancing algorithm is employed here. Fig. 3 gives a sketch of the Hilbert space-filling curve load balancing algorithm. We use the Hilbert curve to link the root cells one by one and create a one-to-one mapping from the cell to the node of the curve as shown in Fig. 3(a). Considering the workload on the node, the curve is cut into n_p (n_p is the number of processors) segments with different sizes. All the segments have similar amount of workload. Corresponding cells on the same

14

segment are assigned to the same processor. As shown in Fig. 3(b), the computational domain is split into a series of non-overlapping portions. Finally, the workload is evenly distributed among all processors.

During the AMR process, refinement grid is often added or removed to regions of the domain. Refinement state is evolving in time which leads to dynamic changing of workload on the node. Thus dynamic load balancing is necessary. The dynamic load balancing is achieved through a process known as domain repartitioning. The domain repartitioning process consists of two steps. First, considering the new refinement state, repartition the domain into n_p sub-domains utilizing Hilbert space-filling curve load balancing algorithm mentioned above. Second, assign sub-domains on processors by migrating data among processors.

(a) Domain for the two-dimensional detonation case partitioned

on 40 processors at three different times

(b) Domain for the three-dimensional detonation case partitioned

on 40 processors
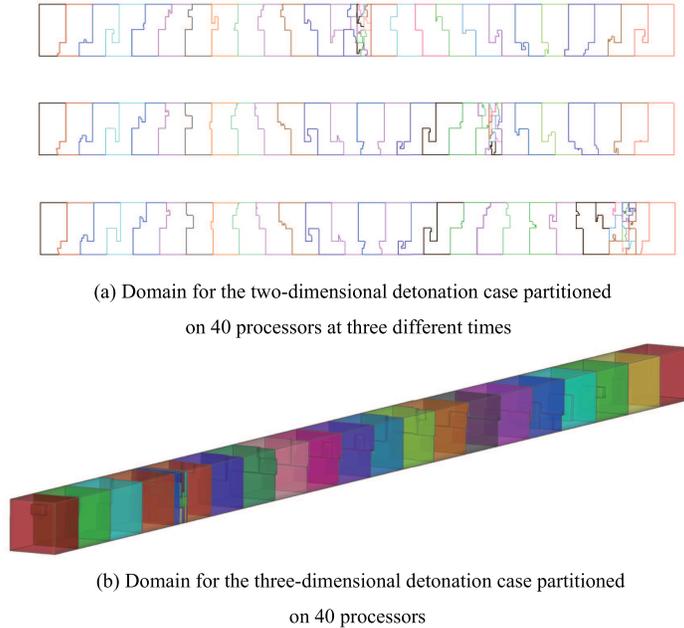
Fig. 4: Domain partition for two dimensional and three dimensional cases

The load imbalance factor metric is used as the criteria for triggering repartition and is defined as follows:

$$LIF_g = \max_{i=1}^{n_p} \left| \frac{W_i}{\sum_{i=1}^{n_p} W_i / n_p} - 1 \right| \tag{19}$$

15

where n_p is the total number of processors in the computation and W_i is the estimation of workload on the i-th processor. The repartition is triggered if $LIF_g > \delta$ with δ being the imbalance tolerance. A typical imbalance tolerance is $\delta = 0.1$, which means that each part must not exceed 10% of the ideal workload. Fig. 4(a) presents the domain partition on 40 processors for the two-dimensional detonation case. It can be found that the domain is dynamic partitioned on processors. Fig. 4(b) shows the domain partition for the three-dimensional detonation case.

4. Numerical results

We have developed the parallel AMR&WENO code based on the method described above and performed simulation on a three dimensional test case to examine the convergence of the method firstly. Then, we move to the study of one, two and three dimensional detonations. We give some conventions here. The refinement ratio in the following simulations is fixed at $r = 3$. Numerical results marked with "AMR&WENO" are obtained utilizing the AMR&WENO code. Results labeled with "WENO" are produced by the classical WENO scheme on a uniform mesh. A relative grid size n is employed to quantify spatial resolution expressed as a reciprocal relationship to the cell size and is defined as follows.

$$n = 5\Delta x/L_{1/2} \tag{20}$$

where $L_{1/2}$ is the half-reaction length which is the distance between the leading shock and the point where half of the reactant is consumed by combustion. Δx is the cell size. In particular, the relative grid size of the AMR&WENO method refers to the relative grid size of the root cells.

4.1. Verification of the numerical resolution

For this test case, the partial differential equations (PDEs) are modified by adding additional source terms so that they accommodate the following exact solution

$$\rho(x, y, z, t) = 1 + \frac{1}{2}\sin(\pi(x + y + z - 3t))$$

$$u(x, y, z, t) = v(x, y, z, t) = \omega(x, y, z, t) = p(x, y, z, t) = 1 \tag{21}$$

$$Y(x, y, z, t) = \frac{1}{2} + \frac{1}{2}\sin(\pi(x + y + z - 3t))$$

16

Tab. 1: Errors and orders of accuracy for density.

Δh	$CFL = 0.2$ error	order
2/10	$1.25E - 02$	
2/20	$6.04E - 04$	4.39
2/30	$7.78E - 05$	5.06
2/40	$1.86E - 05$	4.98
2/50	$6.13E - 06$	4.97

In this case, the PDEs actually become a linear system. The activation energy $E_a = 50.0$, overdrive factor $f = 1.6$, heat of reaction $q = 50.0$, and the specific heat ratio $\gamma = 1.2$ are fixed. The computational domain is taken as $[0, 2] \times [0, 2] \times [0, 2]$. The AMR&WENO is performed on the root cells with $\Delta x = \Delta y = \Delta z = \Delta h$. The refinement regions are placed around the center of the computational domain. The periodic boundary condition is utilized here. Tab. 1 shows the results at time $t = 2.0$. It is obvious that the desired fifth order accuracy is achieved.

4.2. One-dimensional results

The oscillation mechanism of one dimensional unsteady detonation is studied using the WENO&AMR method. The linear stability analysis of the one-dimensional detonation solution indicates that, below a threshold in overdrive, one-dimensional detonation waves in high activation energy are unstable. There are a number of unstable modes which interact with each other in the finite-amplitude, nonlinear regime. In the nonlinear regime, small difference in truncation errors between algorithms eventually leads to dramatic change of solutions [24]. We use the one dimensional unsteady detonation solution to validate the WENO&AMR code.

We consider a long tube with dimension $1000L_{1/2}$. The computational domain is split in 40 portions and distributed in 40 processors. A steady overdriven detonation is located at $x = 200L_{1/2}$ in the initial state. The detonation front is set far from the left side of the tube in order to reduce the influence of the boundary. Here and in the following, the values of the heat Release, activation energy and ratio of specific heats are fixed at $q = 50$, $E_a =$

50 and $\gamma = 1.2$, unless otherwise stated. The constant rate coefficient K is a scaling factor and is fixed at an appropriate value so that the characteristic length scale is the half-reaction length of the steady wave. In this case, the threshold of the overdrive factor is $f = 1.73$. The detonation wave is stable for $f \geq 1.73$ and there is only a single mode of instability for $1.73 > f > 1.57$ [25].

Two test cases are considered here. The first case is the detonation with single mode instability at $f = 1.6$. In this test case, there is an exact result computed from linear stability theory for the basic model owing to Fickett and Wood [26]. Three kinds of grid resolution are considered in this test case in order to check the mesh convergence of AMR&WENO. The Second case is a unstable detonation with the overdrive factor $f = 1.5$.

Fig. 5 shows the history of the peak pressure near the shock front with $f = 1.6$. The curve with square symbols labeled with WENO is obtained utilizing a uniform mesh with $n = 1.5$. The other curve with triangular symbols is obtained by the AMR&WENO code and the relative cell size of the "root cells" is $n = 0.5$. The grid resolution near the reaction zone is therefore the same for both methods. The coincidence of the two curves indicates that the AMR&WENO performance is as good as that of WENO with a relatively lower grid resolution in most part of the computational domain. The amplitude and the period of the oscillation agree well with the results using the uniform grid WENO scheme [27].
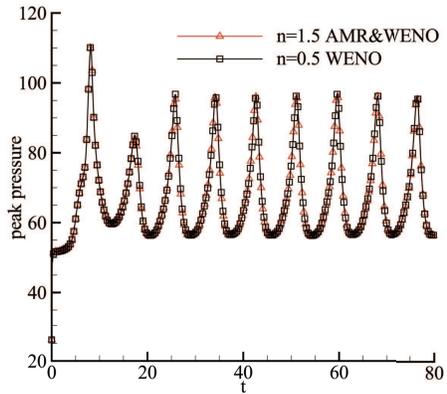

Fig. 5: History of peak pressure with $f = 1.6$.

18

A simulation with $f = 1.6$ is repeated using three kinds of grid resolution. They are $n = 0.75$, $n = 1.5$ and $n = 3$. Fig. 6(a) shows the history of the peak pressure in the shock front and Fig. 6(b) is a convergence comparison with the exact solution between the AMR&WENO method and the WENO scheme with uniform meshes. Fig. 6(a) illustrates a mesh refinement convergence during the period of peak pressure oscillation and Fig. 6(b) illustrates that the peak pressure amplitude of AMR&WENO converges to the exact solution. The AMR&WENO method performance is as good as WENO with a lower grid resolution in most part of the computational domain. The convergence of the AMR&WENO both during the period of oscillation and in the peak pressure amplitude validates the practicability and the effectiveness of AMR&WENO. Fig. 6 also gives a required resolution $n = 1.5$ for the simulation of detonation. In the following test case, the relative grid size is fixed at $n = 1.5$.

The second test case uses AMR&WENO with the grid resolution $n = 1.5$. In this test case, the overdrive is reduced to $f = 1.5$. As reported in Hwang et al. [27], linear stability analysis suggests that there are two unstable modes generated at this overdrive. The history of the shock front pressure shown in Fig. 7 agrees well with the results using a uniform mesh WENO scheme [27].

4.3. Two-dimensional results

Detonation fronts usually have a complicated three-dimensional, time-dependent structure with interior transverse shock waves. There are many investigations on two-dimensional cellular detonations and the transverse wave structures. In this section, the structure of strong transverse waves in two dimensional numerical simulations of detonations is studied.

In the simulation of two-dimensional detonation, the parameters used are $f = 1.0$, $q = 50$, $E_a = 50$ and $\gamma = 1.2$. The test case in two-dimensional detonation is focused on the detailed structure of the strong transverse waves. In this case, detonation wave travels from left to right along a rectangle tube with dimension $10L_{1/2} \times 240L_{1/2}$. The AMR&WENO method is performed on 40 processors. The left boundary of the tube is an inflow, the right boundary is an outflow, and the remaining sides are rigid walls. The initial condition is given by placing the steady, one-dimensional detonation on the domain in the region $x < 10L_{1/2}$. The precursor shock wave is located at $x = 10L_{1/2}$. A perturbation of density is placed in the region of $10L_{1/2} \leq x \leq 11L_{1/2}$ to
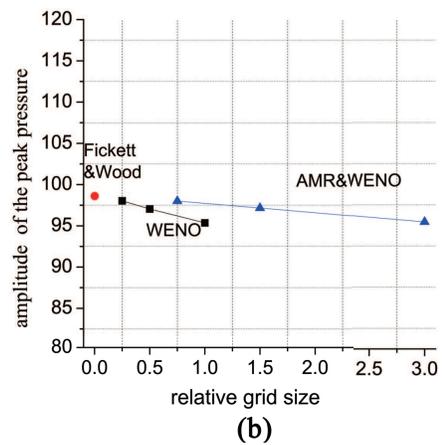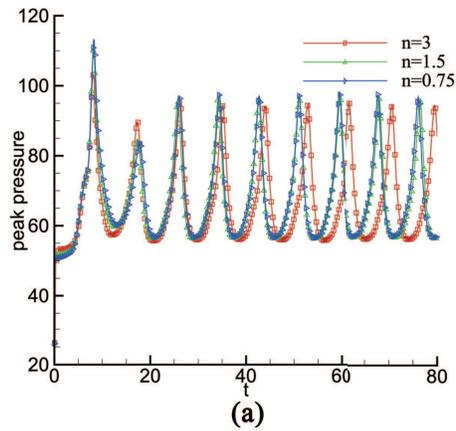
19

(a)

(b)

Fig. 6: Convergence study. (a) History of the peak pressure with differ-ent grid resolution; (b) Amplitude of the peak pressure with different grid resolution.
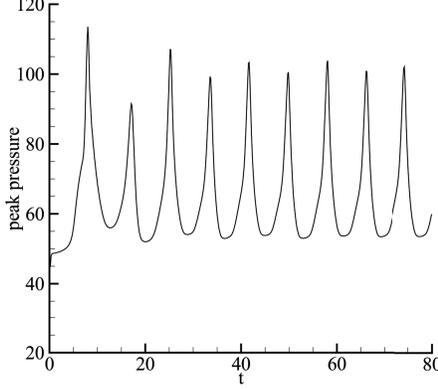
Fig. 7: History of the peak pressure with $f = 1.5$.

initiate a disturbance. The form of the disturbance is as following

$$\rho' = \begin{cases} 0 & , \ x < 10L_{1/2} \\ \dfrac{\sin(\pi(11L_{1/2} - x))}{4} \times [1 + \cos(\dfrac{\pi y}{10L_{1/2}})] & , \ 10L_{1/2} \leq x \leq 11L_{1/2} \\ 0 & , \ x > 10L_{1/2} \end{cases} \quad (22)$$

After giving the initial condition and boundary condition, the simulation is performed using AMR&WENO and WENO schemes. The grid resolution of the WENO scheme is $n = 0.7$. The grid resolution of AMR&WENO in the root cells and the grid resolution of WENO are the same. Since the density and pressure change rapidly across the shock wave and the density has a dramatic change across the contacts and slip lines, we investigate the detailed structure of detonation shown in the pictures of the density gradient $|grad(\rho)|$ and the pressure gradient $|grad(p)|$. The mass fraction Y and the $|\omega|$ reaction rate are used to reveal the reaction structure.

Fig. 8 shows the mesh near the detonation front at a particular time. As mentioned above, the refinement criteria in this picture is a combination of the pressure gradient and the mass fraction gradient. In the process of calculation, the mesh is adaptively refined in the area where the value of the pressure gradient and the mass fraction gradient is relatively large. Figs. 9-12 show the density gradient, pressure gradient, mass fraction and the reaction rate of AMR&WENO and the WENO scheme. For the detonation with

21

strong transverse waves, there will be two triple points in the detonation wave [28]. Take Fig. 9(a) as an example. One triple point which is labeled as $t2$ appears on the front of the precursor shock wave. The incident shock I and Mach stem M which is kinked at the point b intersect at the triple point $t2$. The transverse shock wave T is kinked at the second triple point $t1$ which is inside the reaction zone. This is different from the weak transverse case. Weak transverse extends back into the flow from the triple point on the shock front without being distorted. S is a slip line which extends from the triple point $t2$ and rolls up near the upper boundary. Fig. 10 shows that the pressure gradient of the segment between $t1$ and a is large, which indicates that there is a shock. Fig. 12 shows that the slip line S separates the areas of the completely burnt gas mixture and the unburnt mixture. There is another slip line R far away from the shock front.
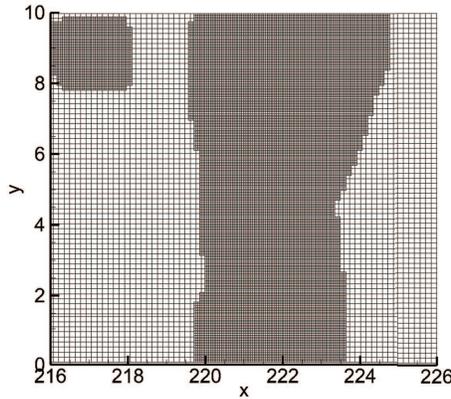
Fig. 8: Mesh near the detonation front.

Although the base grid resolutions of the AMR&WENO and WENO are the same, the results are quite different. The density gradient in Fig. 9(a) is obtained using AMR&WENO and Fig. 9(b) using the WENO scheme. The results of the WENO scheme appear to be in qualitative agreement with that of AMR&WENO. The incident shock I, Mach stem M, kinked point b and triple point on the shock front can be identified in both figures. But Fig. 9(a) shows a higher resolution and shows clearer structures than Fig. 9(b). The result of WENO shows a fuzzy outline of the slip line S and fails to describe the slip line R. AMR&WENO reveals the detailed helical

22

structure of the slip line S and gives the position sharply of the slip line R. The shock between $t1$ and a which is discernible in Fig. 9(a) cannot be seen in Fig. 9(b). That is, WENO fails to identify this shock wave. The pressure gray-scale map in Fig. 10 confirms this conclusion. The results of WENO in Fig. 11 and Fig. 12 show that the pocket of the unburnt mixture behind the slip line R is completely burnt out. WENO fails to describe the flow fired behind the detonation. With the same base grid resolution $n = 1.5$, AMR&WENO scheme gives better resolution comparing with WENO. The detailed structure of the strong transverse wave is correctly identified.
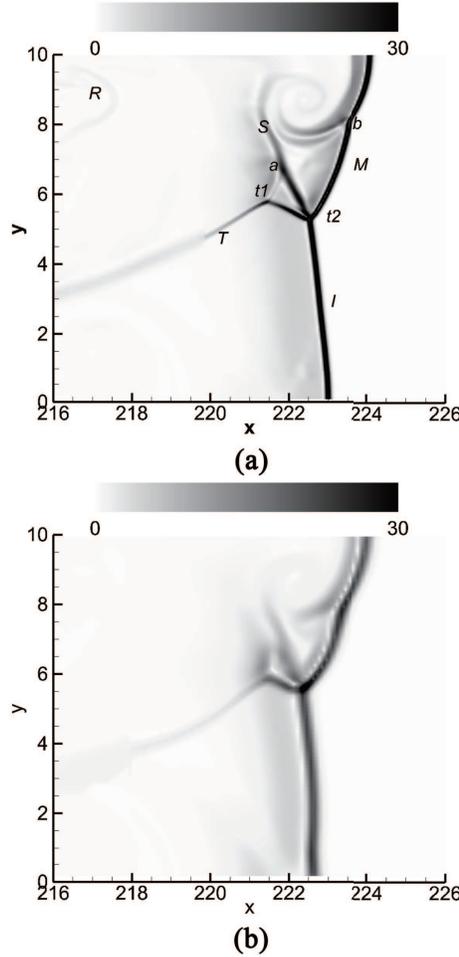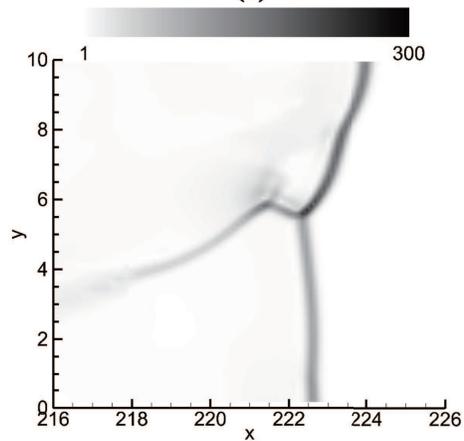

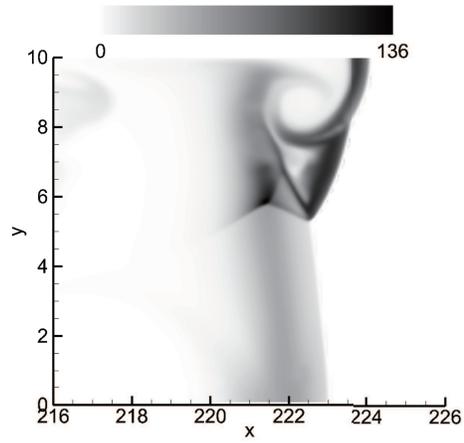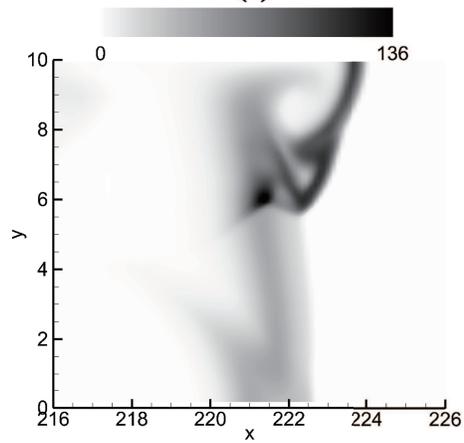Fig. 9: Density gradient. (a) AMR&WENO. (b) WENO.
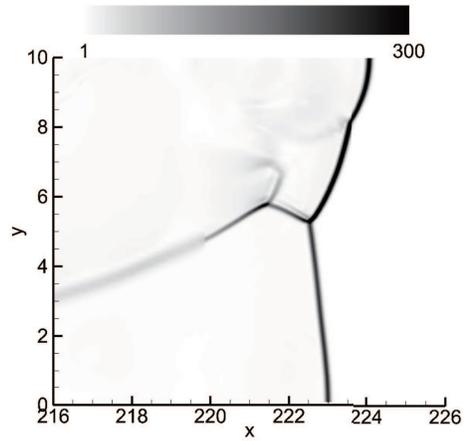
Fig. 10: Pressure gradient. (a) AMR&WENO. (b) WENO.
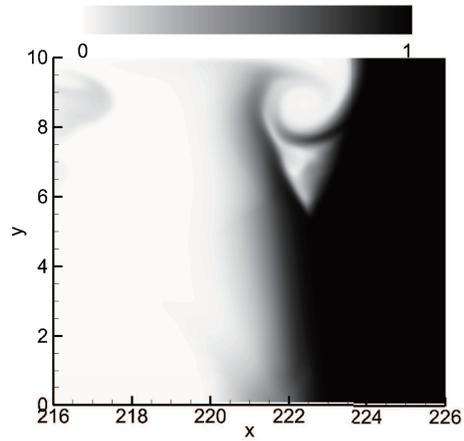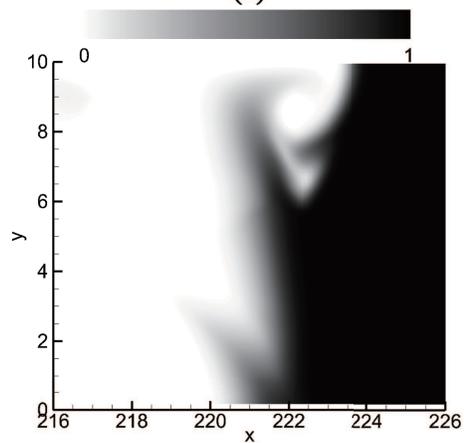
Fig. 11: Reaction rate. (a) AMR&WENO. (b) WENO.

Fig. 12: Mass fraction. (a) AMR&WENO. (b) WENO.

4.4. Three-dimensional results

In the simulation of three dimensional detonation, we consider the detonation wave traveling in a long, straight and rectangular duct. As is well known, the duct width has a great influence on the detonation propagation mode when it is small. For unstable detonation, when the duct width is smaller than a threshold value, the unstable detonation propagates in a spinning mode [13]. In this paper, we set the detonation parameters to $f = 1.0$, $q = 50$, $E_a = 50$ and $\gamma = 1.2$. One-dimensional ZND analytical solution is employed as the initial condition. A very small random perturbation in the ZND profile is added to accelerate the unstable growth of the flow. In order to observe the detonation derivation in a long period of time, we give a relatively long duct length with $240L_{1/2}$. The square duct width is fixed at $2L_{1/2}$. The left boundary of the duct is an inflow, the right boundary is an outflow, and the remaining sides are rigid walls. We use the parallel AMR&WENO code to simulate propagation of detonation. The code is used on a high-performance computer. The same relative grid size $n = 1$ is adopted in all of the following simulations.

Simulation is firstly performed on 40 processors using the AMR&WENO code whose maximum refinement level is $l_{max} = 1$. The simulation results distributed on 40 processors are organized to a complete computational domain. We cut the one part off the global domain where refinement cells are located. The cells on the refinement level 1 form an irregular domain referred as "pool". Fig. 13 shows the density contours of the spinning detonation front. Only the density contours on the "pool" is displayed. We can observe the edge of the "pool" in the picture. Fig. 13(a)-(j) give a series of frames in a typical spinning detonation cycle. The AMR&WENO code produces results that are as good as using a WENO scheme with a relative grid size $n = 0.25$ [13].

The AMR&WENO code allows for multiple refinement to obtain a higher local resolution. In the following work, we simulate the same problem using the AMR&WENO code with the maximal refinement level $l_{\max} = 2$. For comparison purposes, we also do the same simulation utilizing the classical WENO scheme on uniform meshes. Three sets of results are shown in Fig. 14 and Fig. 15. These results are obtained by using WENO, AMR&WENO with maximal refinement level $l_{\max} = 1$ and AMR&WENO with maximal refinement level $l_{\max} = 2$, respectively. Fig. 14(a)-(c) give the meshes of the domain for the three cases. For the WENO scheme case, the initial domain is cut into cells with uniform size as shown in Fig. 14(a). Resolution in the
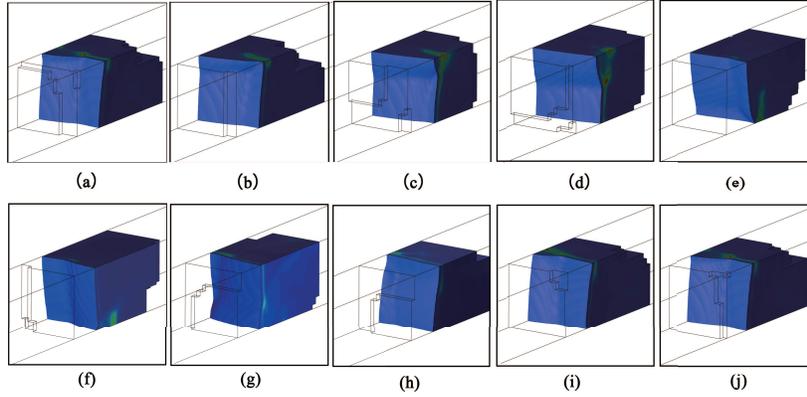
Fig. 13: Density contours of the spinning detonation front on the "pool" of level 1. Frames (a)-(j) show a typical spinning detonation cycle.
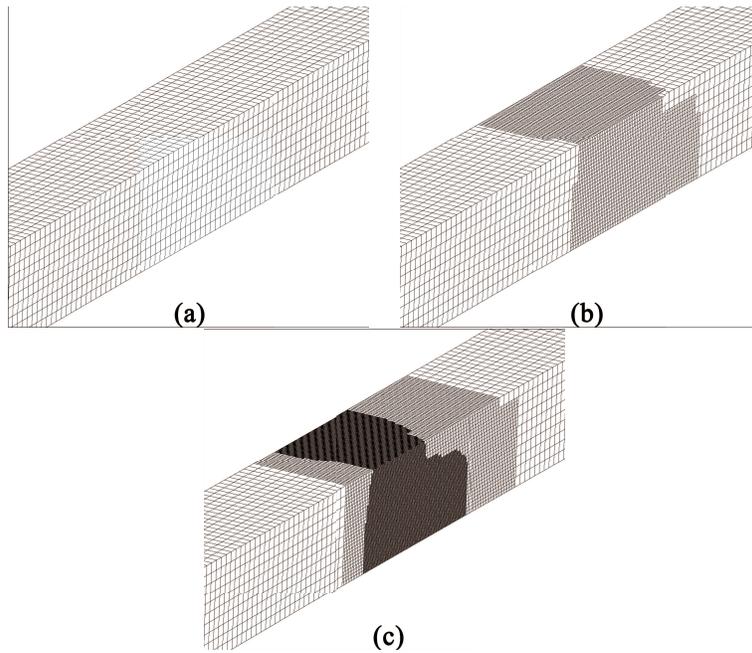

Fig. 14: Mesh near the detonation front. (a) WENO with uniform mesh. (b) AMR&WENO with refinement level $l_{max} = 1$. (c) AMR&WENO with level $l_{max} = 2$.
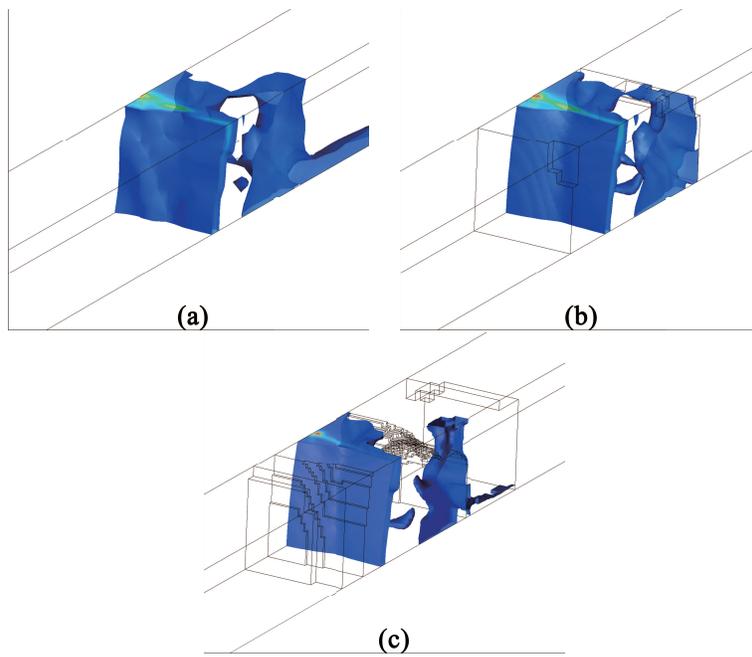
Fig. 15: Density iso-surface of the spinning detonation front. (a) WENO with uniform mesh. (b) AMR&WENO with refinement level $l_{max} = 1$. (c) AMR&WENO with level $l_{max} = 2$.

global domain is the same. For AMR&WENO with $l_{\max} = 1$, the domain is cut into cells with uniform size. The regions where a high local resolution is needed are determined, and cells in these regions are divided into smaller ones as shown in Fig. 14(b). The subdivided cells can be further divided into smaller ones to get higher local resolution as displayed in Fig. 14(c). Fig. 15(a)-(c) give the density iso-surfaces of the spinning detonation front for the three cases. By comparing resolution on the iso-surfaces, we can find that AMR&WENO with $l_{\max} = 2$ has the highest resolution. AMR&WENO with $l_{\max} = 1$ case follows it. Both of them are better than the WENO scheme. The comparison confirms the superior performance of the AMR&WENO method.

5. Conclusions

In this paper, a well-designed adaptive mesh refinement (AMR&WENO) method is proposed to simulate multi-dimensional detonation. A new cell-based data structure is used to organize the data. The new data structure makes it possible for cells to communicate with each other quickly and easily. High order WENO interpolation and Hermite interpolation are used to prolong the data from parent cells to child cells. High order prolongation can improve the accuracy of the AMR&WENO method. A workload balancing parallel AMR&WENO code is developed utilizing the Hilbert space-filling curve algorithm and is used for detonation simulation. The high order accuracy of AMR&WENO is verified using smooth test problems. One dimensional result shows that AMR&WENO is convergent and robust in solving detonation problems. The AMR&WENO method is also performed in two and three dimensional simulations. The comparison results between the classical WENO scheme on uniform meshes and AMR&WENO further demonstrate the high performance the AMR&WENO method.

References

[1] M.J. Berger and J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, Journal of Computational Physics, 53(3) (1984) 484-512.

[2] M.J. Berger and P. Colella, Local adaptive mesh refinement for shock hydrodynamics, Journal of Computational Physics, 82(1) (1989) 64-84.

[3] R. Deiterding, Parallel adaptive simulation of multi-dimensional detonation structures, Ph.D Thesis, University of Cottbus, 2003.

[4] E. Loth, S. Sivier and J. Baum, Adaptive unstructured finite element method for two-dimensional detonation simulations, Shock Waves, 8(1) (1998) 47-53.

[5] H. Ji, F.S. Lien and E. Yee, A new adaptive mesh refinement data structure with an application to detonation, Journal of Computational Physics, 229(23) (2010) 8981-8993.

[6] K. Benkiewicz and K. Hayashi, Two-dimensional numerical simulations of multi-headed detonations in oxygen-aluminum mixtures using an adaptive mesh refinement, Shock Waves, 12(5) (2003) 385-402.

[7] H. Wilkening and T. Huld, An adaptive 3-D CFD solver for modeling explosions on large industrial environmental scales, Combustion Science and Technology, 149(1-6) (1999) 361-387.

[8] J.L. Ziegler, R. Deiterding, J.E. Shepherd and D.I. Pullin, An adaptive high-order hybrid scheme for compressive, viscous flows with detailed chemistry, Journal of Computational Physics, 230(20) (2011) 7598-7630.

[9] H. Ji, F.S. Lien and E. Yee, Numerical simulation of detonation using an adaptive Cartesian cut-cell method combined with a cell-merging technique, Computers & Fluids, 39(6) (2010) 1041-1057.

[10] A.M. Khokhlov and E.S. Oran, Numerical simulation of detonation initiation in a flame brush: the role of hot spots, Combustion and Flame, 119(4) (1999) 400-416.

[11] M. Dumbser, A. Hidalgo and O. Zanotti, High order space time adaptive ADER-WENO finite volume schemes for non-conservative hyperbolic systems, Computer Methods in Applied Mechanics and Engineering, 268 (2014) 359-387.

[12] H.S. Dou, H.M. Tsai and B.C. Khoo, Simulations of detonation wave propagation in rectangular ducts using a three-dimensional WENO scheme, Combustion and Flame, 154(4) (2008) 644-659.

[13] C. Wang, C.-W. Shu, W. Han and J.G. Ning, High resolution WENO simulation of 3D detonation waves. Combustion and Flame, 160(2) (2013) 447-462.

[14] X.Y. Hu, B.C. Khoo, D.L. Zhang and Z.L. Jiang, The cellular structure of a two-dimensional $H_2/O_2/Ar$ detonation wave, Combustion Theory and Modeling, 8(2) (2004) 339-359.

[15] S.T. Li and J.M. Hyman, Adaptive mesh refinement for finite difference WENO schemes, Technical Report LA-UR-03-8927, Los Alamos National Laboratory, 2003.

[16] A. Baeza and P. Mulet, Adaptive mesh refinement techniques for high-order shock capturing schemes for multi-dimensional hydrodynamic simulations, International Journal for Numerical Methods in Fluids, 52(4) (2006) 455-471.

[17] C. Shen, J.M. Qiu and A. Christlieb, Adaptive mesh refinement based on high order finite difference WENO scheme for multi-scale simulations, Journal of Computational Physics, 30(10) (2011) 3780-3802.

[18] G.-S. Jiang and C.-W. Shu, Efficient implementation of weighted ENO schemes, Journal of Computational Physics, 126 (1996) 202-228.

[19] C.-W. Shu and S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, Journal of Computational Physics, 77(2) (1988) 439-471.

[20] A.M. Khokhlov, Fully threaded tree algorithms for adaptive mesh fluid dynamics simulations, Journal of Computational Physics, 143(2) (1998) 543-519.

[21] K. Sebastian and C.-W. Shu, Multidomain WENO finite difference method with interpolation at subdomain interfaces, Journal of Scientific Computing, 19(1-3) (2003) 405-438.

[22] J. Steensland, S. Chandra and M. Parashar, An application centric characterization of domain-based SFC partitioners for parallel SAMR, IEEE Transactions on Parallel and Distributed Systems, 13(12) (2002) 1275-1289.

[23] W.D. Henshaw and D.W. Schwendeman, Parallel computation of three-dimensional flows using overlapping grids with adaptive mesh refinement, Journal of Computacional Physics, 227(16) (2008) 7469-7502.

[24] D.N. Williams, L. Bauwens and E.S. Oran, A numerical study of the mechanisms of self-reignition in low-overdrive detonations, Physics and Astronomy, 6(2) (1996) 93-110.

[25] A. Bourlioux, A.J. Majda and V. Roytburd, Theoretical and numerical structure for unstable one-dimensional detonations, SIAM Journal on Applied Mathematics, 51(2) (1991) 303-343.

[26] W. Fickett and W.W. Wood, Flow calculations for pulsating one-dimensional detonations, Physics of Fluids, 9(5) (1966) 903-916.

[27] P. Hwang, R.P. Fedkiw, B. Merriman, T.D. Aslam, A.R. Karagozian and S.J. Osher. Numerical resolution of pulsating detonation waves, Combustion Theory and Modelling, 4(3) (2000) 217-240.

[28] G.J. Sharpe, Transverse waves in numerical simulations of cellular detonations, Journal of Fluid Mechanics, 447 (2001) 31-51.