

ALTERING CONNECTIVITY WITH LARGE DEFORMATION MESH FOR LAGRANGIAN METHOD AND ITS APPLICATION IN MULTIPLE MATERIAL SIMULATION^{*†}

Ruili Wang^{1‡}; Xiao Liang^{1,2}, Zhong Lin¹

- (1. *Institute of Applied Physics and Computational Math., Beijing 100094, PR China;*
2. *College of Math. and Information Science, Shandong University of Science and Technology, Shandong 266590, PR China*)

Abstract

A new approach for treating the mesh with Lagrangian scheme of finite volume method is presented. It has been proved that classical Lagrangian method is difficult to cope with large deformation in tracking material particles due to severe distortion of cells, and the changing connectivity of the mesh seems especially attractive for solving such issues. The mesh with large deformation based on computational geometry is optimized by using new method. This paper develops a processing system for arbitrary polygonal unstructured grid, the intelligent variable grid neighborhood technologies is utilized to improve the quality of mesh in calculation process, and arbitrary polygonal mesh is used in the Lagrangian finite volume scheme. The performance of the new method is demonstrated through series of numerical examples, and the simulation capability is efficiently presented in coping with the systems with large deformations.

Keywords Lagrangian scheme; large deformations; altering connectivity of mesh; finite volume method

2000 Mathematics Subject Classification 74S10

1 Introduction

The relationship between the distortion of the computational cells and the motion of the fluid plays an important role in numerical simulation of multidimensional compressible flow. Lagrangian or Eulerian coordinate is utilized to resolve this problem. However, there is a great difference between these two methods. For instance,

^{*}The work was supported in part by the National Natural Science Foundation of China under Grant 11372051, Grant 11475029, in part by the Fund of the China Academy of Engineering Physics under Grant 20150202045.

[†]Manuscript received April 21, 2016

[‡]Corresponding author. E-mail: Wang_Ruili@iapcm.ac.cn

the computational grid moves with the local fluid velocity in the Lagrangian description, while the grid is fixed in the Eulerian framework. The Lagrangian methods are prevalent in compressible multi-material fluid dynamics with high temperature and pressure, since this approach has well-resolved material interfaces and does not have advective diffusion term. However a large shearing distortion may lead to severely distorted cells and, inevitably, to mesh tangling, which will reduce the accuracy of the discrete scheme based on the grid, and the computation will run termination. Therefore, to resolve the large deformation is one difficulty and the focus in Lagrangian methods, and also the front field in CFD at present.

The deformation of the Lagrangian methods originates from two sources: One is un-robustness of the numerical scheme, and the other is grid evolution following the fluid. Thus highly qualified grid and robust scheme must be explored in order to make the Lagrangian methods having strong adaptation.

In this paper we propose an automatically local remeshing method based on changing connectivity of the mesh, which can be used to handle geometric intersection. It consists of two parts. First an arbitrary unstructured polygonal mesh is constructed through a collection of control nodes that are topologically organized into cells. By the way, the mesh is unstructured in the sense that individual cells may be constructed from arbitrary, non-uniform number of nodes. Second, changing connectivity of the mesh (topology transformation) is allowed during numerical simulation. Topological operations such as splitting and elimination of cells and edges, merging of cells is allowed in this process. This approach has successfully been implemented in a number of 2D codes of numerical analysis.

2 Governing Equations

Consider a two-dimensional multi-material compressible fluid system with elastic and plastic terms written in Lagrangian formalism given by:

$$\begin{aligned}
 \frac{\dot{V}}{V} &= \dot{\epsilon}_x + \dot{\epsilon}_y + \dot{\epsilon}_\theta, \\
 \rho \frac{du}{dt} &= \frac{\partial \Sigma_x}{\partial x} + \frac{\partial T_{xr}}{\partial r} + \nu \frac{T_{xr}}{r}, \\
 \rho \frac{dv}{dt} &= \frac{\partial T_{xr}}{\partial x} + \frac{\partial \Sigma_r}{\partial r} + \nu \frac{\Sigma_r - \Sigma_\theta}{r}, \\
 \rho \frac{de}{dt} &= \Sigma_x \dot{\epsilon}_x + \Sigma_r \dot{\epsilon}_y + \Sigma_\theta \dot{\epsilon}_z + T_{xr} \dot{\gamma}_{xr},
 \end{aligned} \tag{1}$$

where x and r denote axes. In 2D planar problem, x is the level direction, r is the perpendicular direction. In cylindrical symmetry model, x is the axial symmetry direction, r is the perpendicular direction. u, v, ρ, e and p are x -velocity component,

r -velocity component, density, internal energy and pressure respectively. The upper mark “.” denotes the derivative respect to time t , $V = \frac{1}{\rho}$ is the capacity ratio, $\dot{\epsilon}_x, \dot{\epsilon}_y, \dot{\epsilon}_z$ are positive strain ratios, $\dot{\gamma}_{xr}$ is the shear strain ratio. ν is the geometry coefficient, $\nu = 0$ represents two dimensional planar problem, $\nu = 1$ is for two dimensional cylindrical symmetry model.

The artificial viscosity is imported in order to track the discontinuity near the shock, and it will be turned on only when the strain ratio is negative. We regard it to be the linear combination of Landshoff viscosity and Von Neumann-Richtmyer viscosity.

The Von Neumann-Richtmyer viscosity is in the following form:

$$q_{NR} = \begin{cases} l_{NR}^2 \rho \left(\frac{\dot{V}}{V}\right)^2, & \dot{V} < 0, \\ 0, & \dot{V} \geq 0, \end{cases} \quad (2)$$

where l_{NR} has length dimension, with $l_{NR}^2 = a_{NR}A$ and a_{NR} being N-R viscosity coefficient, A is the area of the grid cell.

The Landshoff viscosity is given by

$$q_L = \begin{cases} l_L \rho c \left(\frac{\dot{V}}{V}\right), & \dot{V} < 0, \\ 0, & \dot{V} \geq 0, \end{cases} \quad (3)$$

where l_L is length, with $l_L = a_L \sqrt{A}$ and a_L being the Landshoff viscosity coefficient.

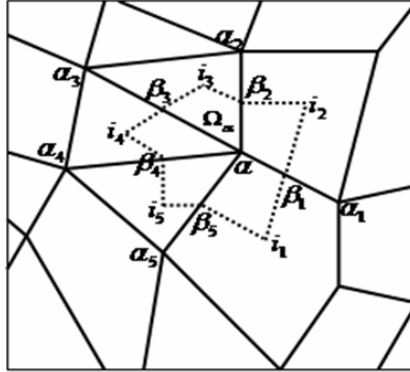
3 Lagrangian Finite Volume on Arbitrary Polygonal Grid

In Lagrangian description, we suppose that edge is the external boundary (natural interface) and the interface of different fluids. Edges, cell and nodes compose the element unit. In this paper, the velocity and coordinates are defined on the nodes, while the other variables (such as density, pressure, stress and strain) are defined on the center of the cell, and they are assumed to be constants over a cell.

The discretion of momentum equation is given in the following:

For each node, e.g. α in Figure 1, we define the control volume to be connected by the center of neighboring cells and the center of edges surrounding the node α , where $\alpha_1, \alpha_2, \dots, \alpha_{m_\alpha}$ are nodes around α , $i_1, i_2, \dots, i_{m_\alpha}$ are the cell centers of the surrounding cells, $\beta_1, \beta_2, \dots, \beta_{m_\alpha}$ are the corresponding centers of the edges.

The discretisation formulations of velocities in α are given through discretion of the momentum equations in the following:


 Figure 1: Control volume Ω_α of momentum equation

$$\begin{aligned}
 u_\alpha^{n+\frac{1}{2}} &= u_\alpha^{n-\frac{1}{2}} + \frac{\Delta t^n}{B_\alpha^n} \sum_{k=1}^{m_\alpha^n} \left\{ [(\Sigma_x)_{\beta_k, i_{k+1}}^n (r_{i_{k+1}}^n - r_{\beta_k}^n) + (\Sigma_x)_{\beta_{k+1}, i_{k+1}}^n (r_{\beta_{k+1}}^n - r_{i_{k+1}}^n)] \right. \\
 &\quad \left. - [(T_{xr})_{\beta_k, i_{k+1}}^n (x_{i_{k+1}}^n - x_{\beta_k}^n) + (T_{xr})_{\beta_{k+1}, i_{k+1}}^n (x_{\beta_{k+1}}^n - x_{i_{k+1}}^n)] \right\} + \nu \Delta t^n F_\alpha^n, \\
 v_\alpha^{n+\frac{1}{2}} &= v_\alpha^{n-\frac{1}{2}} + \frac{\Delta t^n}{B_\alpha^n} \sum_{k=1}^{m_\alpha^n} \left\{ [(T_{xr})_{\beta_k, i_{k+1}}^n (r_{i_{k+1}}^n - r_{\beta_k}^n) + (T_{xr})_{\beta_{k+1}, i_{k+1}}^n (r_{\beta_{k+1}}^n - r_{i_{k+1}}^n)] \right. \\
 &\quad \left. - [(\Sigma_r)_{\beta_k, i_{k+1}}^n (x_{i_{k+1}}^n - x_{\beta_k}^n) + (\Sigma_r)_{\beta_{k+1}, i_{k+1}}^n (x_{\beta_{k+1}}^n - x_{i_{k+1}}^n)] \right\} + \nu \Delta t^n G_\alpha^n,
 \end{aligned} \tag{4}$$

with

$$B_\alpha^n = \sum_{k=1}^{m_\alpha^n} \frac{\rho_{i_k}^n A_{i_k}^n}{l_{i_k}^n}, \quad F_\alpha^n = \frac{2\pi}{m_\alpha^n} \sum_{k=1}^{m_\alpha^n} \frac{(T_{xr})_{i_k}^n A_{i_k}^n}{M_{i_k}}, \quad G_\alpha^n = \frac{2\pi}{m_\alpha^n} \sum_{k=1}^{m_\alpha^n} \frac{(\Sigma_r - \Sigma_\theta)_{i_k}^n A_{i_k}^n}{M_{i_k}}. \tag{5}$$

Using the new velocity in node α , we get the new position of node α for the next time step:

$$x_\alpha^{n+1} = x_\alpha^n + \Delta t^{n+\frac{1}{2}} u_\alpha^{n+\frac{1}{2}}, \quad r_\alpha^{n+1} = r_\alpha^n + \Delta t^{n+\frac{1}{2}} v_\alpha^{n+\frac{1}{2}}.$$

The discretization of the other equations are given in the following section.

For example, the internal energy equation has the following form

$$\dot{\epsilon} = -(p+q)\dot{V} + V(s_x \dot{\epsilon}_x + s_r \dot{\epsilon}_r + s_\theta \dot{\epsilon}_\theta + \tau_{xr} \dot{\gamma}_{xr}). \tag{6}$$

We can discrete (6) on cell i , and use the following sequence to compute step by step

$$\begin{aligned}
 \rho_i^{n+\frac{1}{2}} &= \frac{1}{2}(\rho_i^n + \rho_i^{n+1}), \quad S_i^{n+\frac{1}{2}} = \frac{1}{2}(S_i^n + S_i^{n+1}) \\
 (\Delta Z)_i^{n+\frac{1}{2}} &= \frac{\Delta t^{n+\frac{1}{2}}}{\rho_i^{n+\frac{1}{2}}} (s_x \dot{\epsilon}_x + s_r \dot{\epsilon}_r + s_\theta \dot{\epsilon}_\theta + \tau_{xr} \dot{\gamma}_{xr})_i^{n+\frac{1}{2}},
 \end{aligned} \tag{7}$$

$$\tilde{e}_i = e_i^n + (p_i^n + q_i^{n+\frac{1}{2}}) \frac{\rho_i^{n+1} - \rho_i^n}{\rho_i^{n+1} \rho_i^n} + (\Delta Z)_i^{n+\frac{1}{2}}, \tag{8}$$

$$\tilde{p}_i = f(\tilde{e}_i, \rho_i^{n+1}),$$

finally we get

$$e_i^{n+1} = \tilde{e}_i + \frac{1}{2}(\tilde{p}_i - p_i^n) \frac{\rho_i^{n+1} - \rho_i^n}{\rho_i^{n+1} \rho_i^n} \tag{9}$$

with

$$p_i^{n+1} = p(e_i^{n+1}, \rho_i^{n+1}). \tag{10}$$

4 Polygonal Meshes and Motivation

Consider a 2D domain Ω , which is assumed to be a general polygon. We assume that the cells of the mesh, $\{c\}$, cover the domain Ω without gaps or overlaps. Each cell may be a general polygon, and is assigned a unique index. For simplicity, it is also written by c . The set of vertices (nodes) of the polygons is denoted by $\{n\}$, where each node has a unique index n . Then each cell can be defined by an ordered set of vertices. We denote the set of vertices of a particular cell c by $N(c)$. Further, we denote the set of cells that share a particular vertex n by $C(n)$. Note that each vertex may be shared by an arbitrary number of cells, and we denote connectivities of the cell and node by $CNL(n)$ and $NCL(c)$ respectively. The connectivity of the cell ($CNL(n)$) is defined as an ordered set of nodes of around the cell, the connectivity of the node ($NCL(c)$) is defined as an ordered set of cells of sharing the node (see Figure 2).

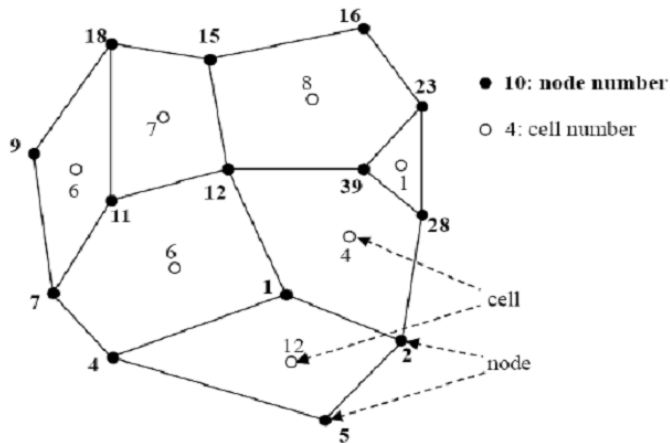


Figure 2: Polygonal mesh and notations

Set of nodes for cell $c = 12$ is $CNL(c = 12) = \{5, 2, 1, 4\}$, and the set of cells sharing node $n = 39$ is $NCL(n = 39) = \{1, 8, 4\}$.

To motivate our research, considering a situation with large deformations, quality of cells degrades and may become unacceptable for further calculation. For instance, the boundaries of Ω significantly changes as time progresses in numerical simulation of multidimensional fluid flow, such that the boundaries may become topologically disconnect in some of its region or singularities may be removed on the boundary. Here “singularities” means thin, wedge-shaped boundary cells, or intersection of cells and edges. If connectivity of the mesh can be changed in unacceptable position, then the general polygonal mesh may be generated in this case. A quadrilateral cell can be remeshed by using topological operations such as elimination of cells and edges to triangles. This stage will be described in Section 5.

5 Topology Transformations

These transformations include rearrangement of the local topological structure in order to arrive at a condition that will better admit a topology to form good elements. Some of these transformations have been described in other literatures, and they will be used here to handle geometric intersection algorithm. Each of the following transformations implements this research, and the basic transformations considered here are as follows.

5.1 Distinguishing intersection

If we implement topological operations for poor quality and unacceptable element, two notions will be given in advance. Firstly, the concave-point in cell will be distinguished by using vector product

$$\begin{aligned} (R_{\alpha_j} - R_{\alpha_{j-1}}) \times (R_{\alpha_{j+1}} - R_{\alpha_j}) \cdot k &= |R_{\alpha_j} - R_{\alpha_{j-1}}| |R_{\alpha_{j+1}} - R_{\alpha_j}| \sin \phi \\ &= \det \begin{pmatrix} x_{\alpha_j} - x_{\alpha_{j-1}} & r_{\alpha_j} - r_{\alpha_{j-1}} & 0 \\ x_{\alpha_{j+1}} - x_{\alpha_j} & r_{\alpha_{j+1}} - r_{\alpha_j} & 0 \\ 0 & 0 & 1 \end{pmatrix}, \end{aligned}$$

where k denotes unit vector in θ -direction, in 3D space $(x - r - \theta)$, $k = (0, 0, 1)^T$. ϕ is the angle between the vectors $(R_{\alpha_j} - R_{\alpha_{j-1}})$ and $(R_{\alpha_{j+1}} - R_{\alpha_j})$.

If $(R_{\alpha_j} - R_{\alpha_{j-1}}) \times (R_{\alpha_{j+1}} - R_{\alpha_j}) \cdot k < 0$, then the node α_j is a concave-point in mesh- i . In Figure 3(a) the node α_j is a concave-point in mesh- i ; In Figure 3(b) the node α_j is not a concave-point in mesh- i , which is a convex-point in mesh- i .

Then intersection of two lines will be distinguished. In Figure 4, if formulas

$$(R_{\alpha_j} - R_{\alpha_k}) \times (R_{\alpha_{k+1}} - R_{\alpha_k}) \cdot k \geq 0, \tag{11}$$

$$(R_{\alpha_{j-1}} - R_{\alpha_k}) \times (R_{\alpha_{k+1}} - R_{\alpha_k}) \cdot k \leq 0, \tag{12}$$

$$(R_{\alpha_k} - R_{\alpha_{j-1}}) \times (R_{\alpha_j} - R_{\alpha_{j-1}}) \cdot k \geq 0, \tag{13}$$

$$(R_{\alpha_{k+1}} - R_{\alpha_{j-1}}) \times (R_{\alpha_j} - R_{\alpha_{j-1}}) \cdot k \leq 0 \tag{14}$$

are all satisfied, then $\alpha_{j-1} \rightarrow \alpha_j$ and $\alpha_k \rightarrow \alpha_{k+1}$ intersect.

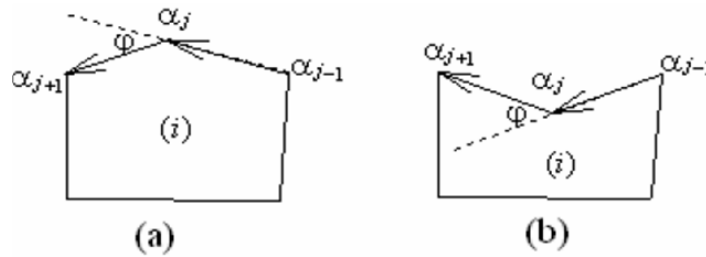


Figure 3: Concave-point and convex-point

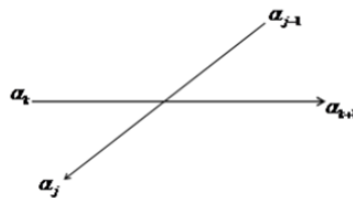


Figure 4: Intersection of two lines

It is easy to distinguish intersection for fragment of the mesh in computing procedure. In Figure 5, the concave-point α_j at t^{n+1} cuts across the edge $\alpha_k \rightarrow \alpha_{k+1}$ in mesh- i , then the mesh- i yields intersection. From this figure, it can be seen that logically mesh- i presents intersection, which cannot be handled during numerical simulation leading to reduce the accuracy of the discrete scheme or the computation will run termination

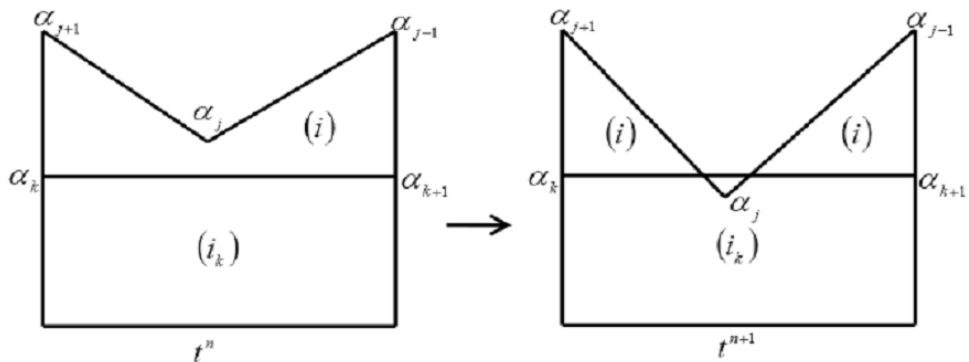


Figure 5: Intersection of two meshes

5.2 Changing connectivity of the mesh techniques

There is a very simple algorithm to handle intersection for fragment of the mesh. This problem can be resolved by using local changing connectivity of the mesh techniques (“topological” remeshing). “Topological” remeshing may be based on topological operations such as splitting and elimination of cells and edges, merging of cells, optimization of poor topological quality mesh and further transformation of unacceptable cells into good cells. Based on unstructured arbitrary polygonal meshes, connectivity of the mesh changed in numerical simulation are defined as lots of topological operations in our LAD2D code [2], such as cutting down edge to the mesh- i on the left or on the right, big cutting down edge to the mesh- i on the left or on the right, merging or refining to the mesh- i , fracture on the node, normal fracture on the common edge of two cells, and so on.

In Figure 6, when the node α_k is equal to the node α_{j+1} ($\alpha_k = \alpha_{j+1}$) of the mesh- i at t^{n+1} , the concave-point α_j cuts across the edge $\alpha_k \rightarrow \alpha_{k+1}$ in mesh- i (only neighbor edge $\alpha_k \rightarrow \alpha_{k+1}$ with α_k), the connectivity of the mesh- i is changed, and eliminates the edge $\alpha_{j+1} \rightarrow \alpha_j$ of the mesh- i , which is defined as cutting down edge to the mesh- i on the left. From Figure 6, it is clear that topological operations with logically quadrilateral element are changed to those with triangles element.

In Figure 7, when the node α_k is equal to the node α_{j+1} ($\alpha_k = \alpha_{j+1}$) of the mesh- i

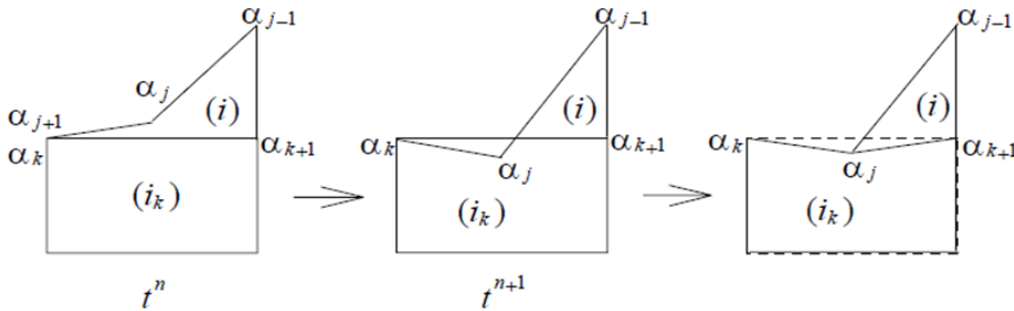


Figure 6: Cutting down edge to the mesh- i on the left

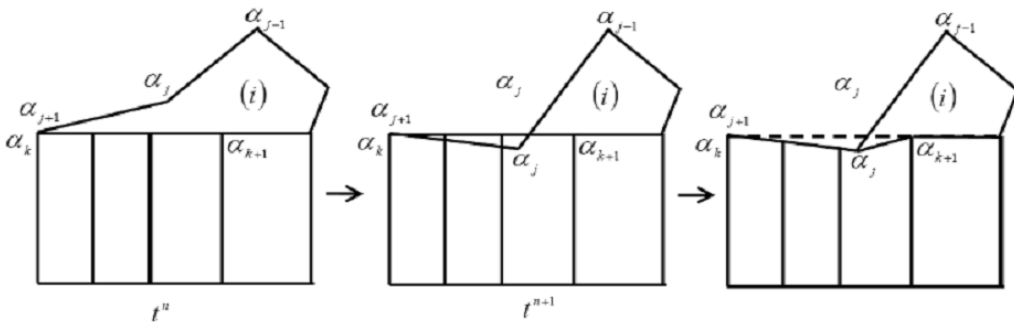


Figure 7: Big cutting down edge to the mesh- i on the left

at t^{n+1} , the concave-point α_j cuts across the edge $\alpha_k \rightarrow \alpha_{k+1}$ in the mesh- i (the other neighbor edge $\alpha_k \rightarrow \alpha_{k+1}$ with α_k), the connectivity of the mesh- i is changed, and eliminates the edge $\alpha_{j+1} \rightarrow \alpha_j$ of the mesh- i , which is defined as big cutting down edge to the mesh- i on the left.

In Figure 8, when the node α_{k+1} is equal to the node α_{j-1} ($\alpha_{k+1} = \alpha_{j-1}$) of the mesh- i at t^{n+1} , the concave-point α_j cuts across the edge $\alpha_k \rightarrow \alpha_{k+1}$ in the mesh- i (only neighbor edge $\alpha_k \rightarrow \alpha_{k+1}$ with α_k), the connectivity of the mesh- i is changed, and eliminates the edge $\alpha_{j-1} \rightarrow \alpha_j$ of the mesh- i , which is defined as cutting down edge to the mesh- i on the right.

In Figure 9, when the concave-point α_j at t^{n+1} cuts across the edge $\alpha_k \rightarrow \alpha_{k+1}$ in the mesh- i , the connectivity of the mesh- i is changed, and eliminates the edge $\alpha_{k+1} \rightarrow \alpha_k$ of the mesh- i , which is defined as merging to the mesh- i , and when refining the edge $\alpha_{k+1} \rightarrow \alpha_k$ to the mesh- i , is defined as refining to the mesh- i .

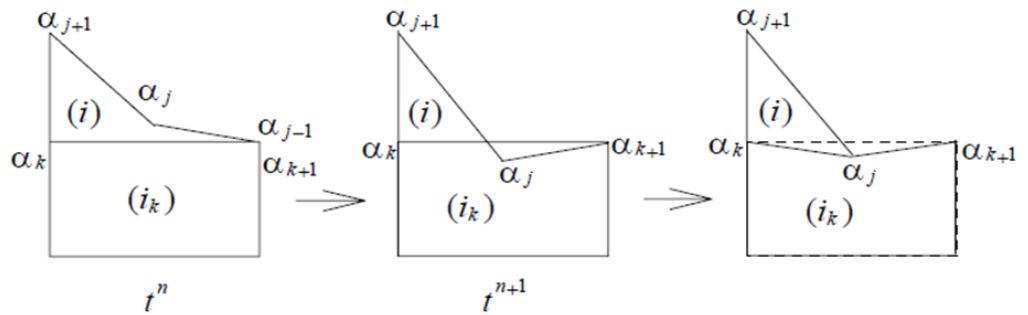


Figure 8: Cutting down edge to the mesh- i on the right

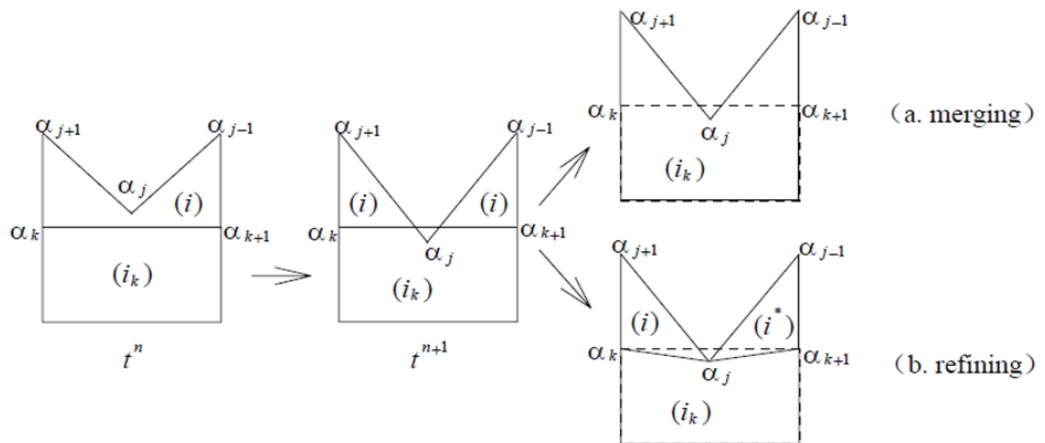


Figure 9: Merging or refining to the mesh- i

The proposed method of automatic local remeshing is implemented as a dynamic link library. In case of remeshing, the calling program receives information on all new removed and changed nodes and cells, and also on intersections of new and old cells which are needed to be rezoned.

After topological operations, Lagrangian solution is transferred (conservatively interpolated) from some polygonal meshes to another.

6 Numerical Tests

In this section, we present the numerical results obtained by the Lagrangian adaptive hydrodynamics code in 2-D space (LAD2D) code [2]. LAD2D is a software project aiming at large scale scientific computing, which is developed and supported by IAPCM. This program is implemented at Yinhe-6 by IAPCM.

The first test is a simple test in planar geometry. It is designed by ourselves and involves break of the air-region with computing procedure. Most of the previous methods are suited for simulating this problem. The second test is “Diffraction of a detonation wave behind a backward-facing step” in $r - z$ -ax symmetric geometry. Numerical simulation is used to validate the predictive capabilities of the new method by comparing with experimental data.

6.1 The simple test case

Consider a simple problem with breaking of region in planar geometry. It is a two layers material problem. The computational domain is $\Omega = [0, 1.0] \times [0, 0.12]$ as described in Figure 10. The lower layer is ideal gas. The gas-region is $\Omega_1^{air} = [0, 1.0] \times [0, 0.10]$. The upper layer is an iron-metal (Fe), the region is $\Omega_2^{Fe} = [0, 1.0] \times [0.10, 0.12]$, the initial densities are $\rho_1 = 0.0129$, $\rho_2 = 7.85$, the initial pressures are $p_1 = 0.1$, $p_2 = 1.0$, the initial velocity is zero in gas-region, the initial velocity is a distribution in the Fe-region as described by formula:

$$v = \begin{cases} 20.0 * \sqrt{(0.9 - x)^2 + (0.1 - y)^2}, & 0.1 < x \leq 0.5, \\ 20.0 * \sqrt{(0.1 - x)^2 + (0.1 - y)^2}, & 0.5 < x \leq 0.9, \\ u = 0. \end{cases}$$

The ideal gas EOS $p = (\gamma - 1)\rho e$ is used in this example with $\gamma = 1.4$. The EOS for iron-metal (Fe) is $p = (\gamma - 1)\rho e + c_0^2(\rho - \rho_0)$, with $\gamma = 4.075$, $\rho_0 = 7.85\text{g/cm}^3$, $c_0 = 4.2\text{km/s}$.

Figure 11 shows the process of automatic separating in gas layer. At the beginning, the iron-metal falls until it meets the boundary of gas, then the gas region is separated into two parts by using automatic local remeshing techniques. From the figure, it is clear that the connectivity of the mesh is changed.

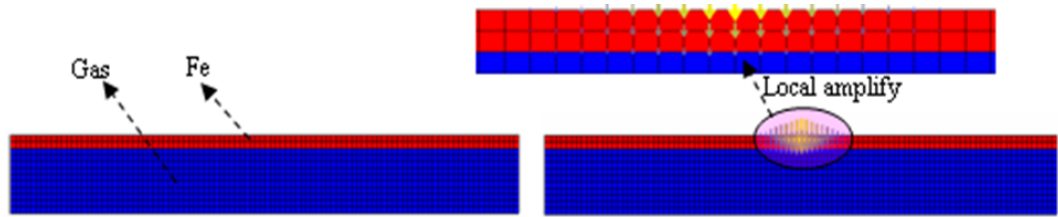


Figure 10: Schematics and meshes of model with initial condition

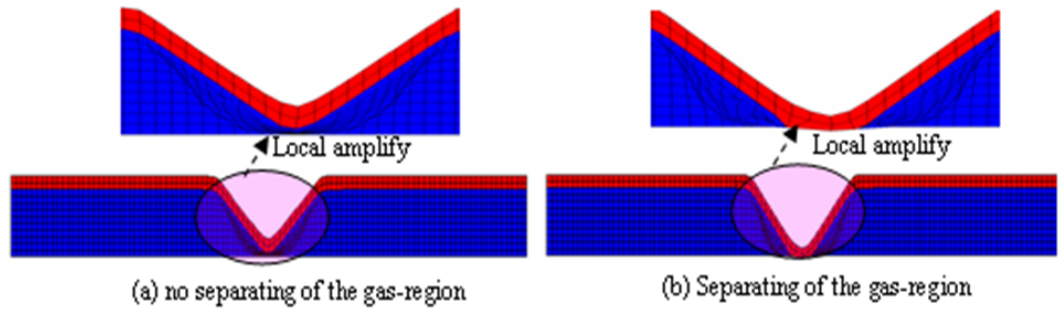


Figure 11: The process of automatic separating in gas layer

6.2 Diffraction of a detonation wave behind a backward-facing step

Diffraction of a detonation wave behind a backward-facing step is one of the fundamental topics in shock wave dynamics and is studied extensively by many researchers. The experiment is used in order to validate the predictive capabilities of the new automatic local remeshing method, based on Lagrangian methods, for problems with large deformations.

This model characterizes the detonation wave propagating through the channels with suddenly expanding section. The left is a little section channel, and the right region is a large section channel. Initially, the region between the inlet and the end of the reaction zone is taken at the CJ state. The walls of the channel holds rigidly, and a no-reflection boundary condition is applied at the outlet. The computational domain Ω is described in Figure 12. From this picture, we can see that Ω is split into two regions filled with the explosive PBX9404 with parameters $K = 2.996$, $\rho_0 = 1.84\text{g/cm}^3$, $D_J = 8.88\text{km/s}$. The left region is $\Omega_1 = [0, 3.0] \times [0, 0.5]$. The right region is $\Omega_2 = [3.0, 6.0] \times [0, 3.0]$. Ω_1 is the driver section, where the top boundary condition satisfies a rigid wall condition and the bottom boundary condition is an axially symmetric.

6.2.1 Reaction rate law

In the numerical simulation of detonation, combustion function F means the extent of explosive reaction. Three zones are distinctly distinguished (see Figure 13).

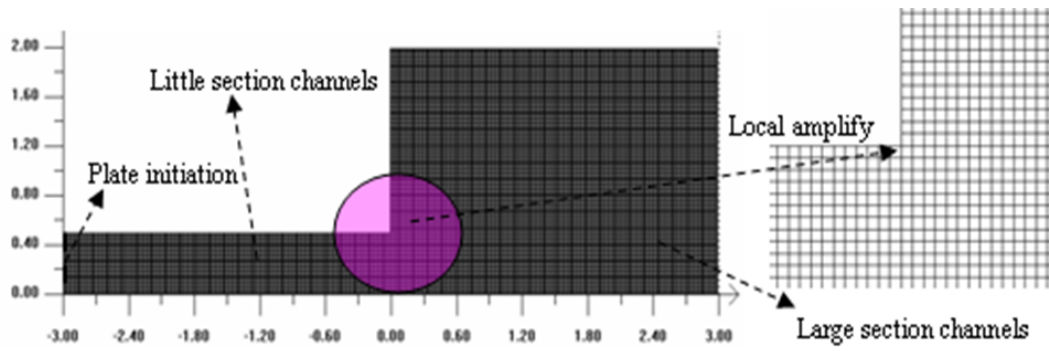


Figure 12: The computational model with initial meshes

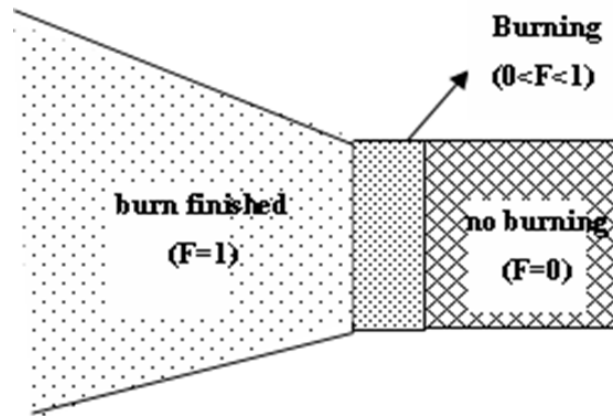


Figure 13: Process of explosive reaction with the combustion function F

The combustion function F is defined as

$$F = [\max\{F_1, F_2\}]^{n_b}, \quad (15)$$

where n_b is the constant to be defined, the Wilkins function F_1 is

$$F_1 = \begin{cases} 0, & V \geq V_0, \\ \frac{V_0 - V}{V_0 - V_J}, & V_0 > V > V_J, \\ 1, & V \leq V_J, \end{cases}$$

and C-J burning function F_2 is defined as

$$F_2 = \begin{cases} 0, & t \geq t_b, \\ \frac{t - t_b}{\Delta L}, & \Delta L + t_b > t > t_b, \\ 1, & t \geq t_b + \Delta L, \end{cases}$$

where $V_J = \frac{\gamma}{\gamma+1}$ is the Chapman-Jouguet relative volume, t is the actual time, t_b is the time for a zone to start burning, $\Delta L = r_b \Delta R / D_J$, ΔR is the cell width, D_J is the detonation velocity, n_b, r_b are adjustable parameters.

6.2.2 Equation of state

The Jones-Wilkins-Lee (JWL) EOS is used to describe the detonation products of the explosive, which has the following form:

$$p = A \exp(-R_1 V) + B \exp(-R_2 V) + \frac{\omega E}{V},$$

where $V = \frac{\rho_0}{\rho}$, $E = \rho_0 e$, the parameters of JWL EOS for explosives are chosen as $A = 852.4 \text{ GPa}$, $B = 18.02 \text{ GPa}$, $R_1 = 4.6$, $R_2 = 1.3$, $\omega = 0.38$.

6.2.3 Results and discussion

Firstly, a steady CJ detonation wave propagates through the narrow segment unit diffraction around a 90° corner, and then begins its travel through the narrow channel as a steady, undisturbed wave. It first senses a change in geometry upon arriving at the corner. As the detonation wave rounds the corner, the diffraction and re-initiation of detonations behind a backward-facing step generate spherical detonation wave. The detonation wave is weakened to deflagration partly in initial the large section channels according to detonation wave diffraction. The slip line could be formed, and the Mach reflection of detonation wave occurs when the detonation wave interacts with the wall. As time elapses, near the line of symmetry (bottom boundary of the computational domain), the wave interacts with the back-face of the step where a vortex exists, the wave becomes planar, the reflection on the wall transits to a Mach reflection and the reflected wave reflects off the upper wall again. Figure 14 shows the mesh in the 90° corner. From this figure it is clear that topolo-

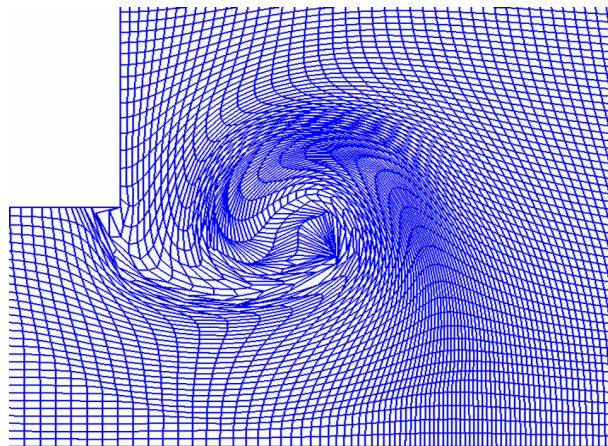


Figure 14: Meshes used in the numerical simulation

gical operation with logically quadrilateral element is changed to that with triangles element. The mesh is so high deformation in near the corner that pure Lagrangian schemes eventually fail.

Figure 15 shows the computing mesh (upper) and density contours (lower) at three times. Diffraction through the 90° corner also generates a stronger corner vortex. We compare numerical results with experimental data obtained from high-speed schlieren photography [6], which coincides with each other qualitatively.

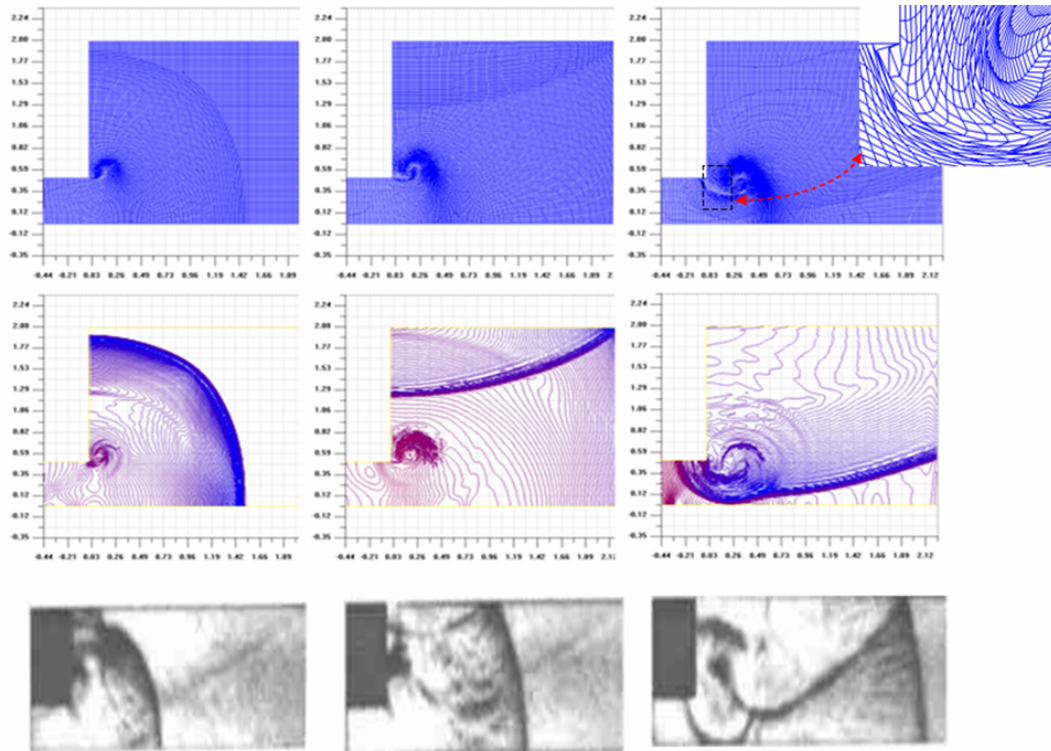


Figure 15: Comparison between the numerical results and the experimental data at different times (Top: Grid used for solving the detonation diffraction; Center: Density contour from calculated result; Bottom: Data from the schlieren photography)

7 Conclusion

In this paper, we present a new approach for treating the mesh in Lagrangian framework based on finite volume method. An unstructured arbitrary polygonal mesh system is defined, and a Lagrangian phase with finite volume methods is obtained by which the solution on polygonal mesh is updated. The automatic local remeshing method is defined using topology transformations techniques. It allows the change of mesh connectivity, and Lagrangian solution is transferred (conserva-

tively interpolated) from some polygonal meshes to another. On our numerical examples, the method is more robust in problems with large deformations, the predictive capabilities of the new automatic local remeshing method is showed. We will focus on the test of this method in the future.

References

- [1] D. Benson, Computational methods in Lagrangian and Eulerian hydrocodes, *Comput. Methods Appl. Mech. Eng.*, **99**(1992),235.
- [2] Ruili Wang, Zhong Lin, Wanzhi Wen, et al, Development and application of adaptive multi-media Lagrangian fluid dynamics software LAD2D, *Computer Aided Engineering*, **23**:2(2014),1-7. (in Chinese)
- [3] J.C. Campbell, M.J. Shashkov, A compatible Lagrangian hydrodynamics algorithm for unstructured grids, *Selcuk J. Appl. Math.*, **4**:2(2003),53-70.
- [4] P. Knupp, L.G. Margolin, and M. Shashkov, Reference Jacobin optimization-based rezoned strategies for arbitrary Lagrangian Eulerian methods, *J. Comput. Phys.*, **176**(2002),93-128.
- [5] Alexander V. Skovpen, Modified algorithm for unstructured quadrilateral meshing, Snezhinsk, RFNC- VNIITF, September 2004, ISBN 5-902278-07-4, Revised on-line version, April 2005.
- [6] S. Ohyagi, T. Obara, S. Hoshi, P. Cai, T. Yoshihashi, Diffraction and re-initiation of detonations behind a backward-facing step, Proceedings of the 18th Int. Colloquium on the Dynamics of explosions and reactive systems at Seattle, USA, 2002.
- [7] Yang Du, Wei Shen, Jian-zhong Zhou, Numerical simulation of detonation wave propagating through the channels with suddenly expending section, *Explosion and Shock Waves*, **24**:1(2004),75-79. (in Chinese)
- [8] R. Wang, Q. Liu, W. Wen, Uncertainty quantification for JWL EOS parameters using non-intrusive polynomial chaos, *Explosive and Shocks*, **35**:1(2015),9-15. (in Chinese)
- [9] R. Loubere, P.H. Maire, M. Shashkov, J. Breil, S. Galera, ReALE: A Reconnection-based arbitrary- Lagrangian -Eulerian method, *Journal of Computational Physics*, **229**(2010),4724-4761.
- [10] Lori A. Freitag, Carl Ollivier-Gooch, Tetrahedral mesh improvement using swapping and smoothing, *International Journal for Numerical Methods in Engineering*, **40**(1997),3979-4002.

(edited by Liangwei Huang)