

文章编号: 1000-0887(2004) 03-0279-12

运动界面的追踪和重构方法^{*}

刘儒勋^{1,2}, 刘晓平², 张磊^{1,3}, 王志峰¹

- (1. 中国科学技术大学 数学系, 合肥 230026;
2. 中国科学技术大学 火灾科学国家重点实验室, 合肥 230027;
3. 中国矿业大学(北京校区) 基础科学学院, 北京 100083)

(戴世强推荐)

摘要: 提出了一种简单有效并且精确的高分辨率运动界面追踪方法——无结构网格特征型积分平均有限体积法, 并且对包括该方法在内的六种界面重构方法做了一些数值试验和评估。通过精确的数值模拟, 详细的比较和分析了这些方法的优缺点。

关键词: 运动界面; 界面重构; 流体体积函数; 积分平均; 有限体积法
中图分类号: O351.1 **文献标识码:** A

引 言

当前, 运动界面问题的数值计算和模拟(包括移动边界或 Stefan 问题, 自由面问题)在自然科学和工程计算中成为越来越热门的课题, 并承担着越来越重要的任务。原因是它涉及到了各种理论和技术领域的数值研究和应用, 例如高等物理和力学, 化学和生物工程, 材料处理和造型等。对液体内气泡的聚合变化和溢出, 晶体的生成和发展, 火焰的闪烁和传播, 金属浇铸和注塑等方面进行数值模拟, 对于理解和研究许多物理现象具有重大的意义。特别是对于这些物理现象中内部和微观的结构和特征的发展过程的数值研究, 具有更加实用的意义。

在过去的 20 年里, 针对运动界面问题的计算和模拟, 已经提出了著名的 VOF (Volume of Fluids) 方法^[1~11]和 Level Set 方法^[12~16]。这些方法在实际应用中取得极好的数值效果。并且通过数值模拟得到了许多极好的关于运动界面的描述。此外, 许多方法已经被编辑成应用软件(诸如 FLOW3D, NASA_VOF, SOLA_VOF, 等等)并应用于半导体处理, 热涂料喷射工艺, 材料加工和科学研究中。

有几种界面捕捉方法, 如 ENO 或 Weighted ENO, 以及间断 Galerkin 方法, 能够和本文提出的积分平均型有限体积法一样, 很好的处理运动界面的问题。但是对于这些界面捕捉方法来说, 仍然需要运动界面的重构技术。

在运动界面追踪中, 由于数值方法的数值耗散和色散效应, 运动界面的轮廓经常会变得模

* 收稿日期: 2002_05_26; 修订日期: 2003_10_06

基金项目: 国家自然科学基金资助项目(10071083); 中国科学技术大学火灾科学国家重点实验室知识创新工程基金资助项目

作者简介: 刘儒勋(1938—), 男, 山东青岛人, 教授, 博导(E-mail: liurx@ustc.edu.cn);
张磊(联系人), Tel: 86_10_62331383, 86_13661003860; E-mail: Zhangl@mail.ustc.edu.cn*

糊或者产生振荡。通过及时的界面重构能够保证运动界面的轮廓精确和锐利。

当前有很多运动界面重构方法。1965年, Harlow 和 Welch 提出了 MAC(Marker and Cell) 方法, 这一方法利用没有质量、动量和能量的标记点, 来表示、追踪和重构流体的运动界面、特别是不可压缩流体的自由面。Hirt 和 Nichols^[1] 针对界面追踪的一般方法以及通过装配格子内界面的斜率来确定运动界面的斜率和位置, 提出了 VOF 方法。在这以后, Ashgriz 和 Poo^[2], Kim^[5] 等人通过对格子内界面斜率更加精确的构造得到了其他不同的重构方法, 尽管他们的精确性和形式是不同的。Rider 和 Kothe^[6], Gueyffier 和 Li jie^[7], Youngs^[8] 等人利用了运动界面的法向矢量近似值。Rudman^[3-4], Vincent 和 Caltagirone^[10], Ubbink 和 Issa^[11] 等人采用了更精细的方法求解流体体积函数方程。

本文提出了一种有效的捕捉方法——特征型积分平均有限体积法。给出了此方法的分析和评估。特别的, 给出了通过严格数值模拟得到的结果。

在接下来的章节安排如下: 第 1 节, 介绍流体体积函数及其相关概念。第 2 节, 给出了一种特征型积分平均有限体积法。第 3 节, 给出了方法的数值试验和结果分析, 并且与其它有效的方法作了比较, 例如 FCT_VOF, Youngs_VOF 方法, FLAIR_VOF 和 Ubbink 和 Issa 的 CICSAM 方法。最后在附录中, 讨论了几种有效的运动界面重构方法。

1 流体体积函数的控制方程

为了追踪运动界面并适时的对运动界面进行重构, 首先给出流体体积函数的控制方程。

1.1 流体体积函数及其控制方程

在一个充满速度为 $V = (u, v)$ 的介质的矢量场中, 定义一个体积为 τ_0 的拉格朗日控制元 σ , 在 σ 内的任何点定义染色体函数

$$\alpha(X) = \begin{cases} 1 & (\text{在 } X \text{ 点上有目标流体}), \\ 0 & (\text{在 } X \text{ 点上无目标流体}). \end{cases}$$

在控制元上, 目标流体的体积为

$$\tau(\sigma) = \int_{\sigma} \alpha(X) dX.$$

由拉格朗日流体体积的特性, 我们有

$$d\tau/dt = 0 \text{ 或 } \partial\tau/\partial t + V \cdot \nabla\tau = 0.$$

其控制元体积 τ_0 为

$$\tau_0 = \int_{\sigma} dX = \text{const}.$$

我们定义流体体积函数, 即目标流体在控制元中所占的比值

$$C = \tau(\sigma)/\tau_0.$$

于是我们得到流体体积函数的控制方程

$$dC/dt = 0 \text{ 或 } \partial C/\partial t + V \cdot \nabla C = 0, \quad (1)$$

如果流体是不可压缩的, 由不可压条件 $\nabla \cdot V = 0$, (1) 式又可写成守恒形式

$$\partial C/\partial t + \nabla \cdot (VC) = 0. \quad (2)$$

由于控制元 σ 是任意的, 因此等式(1)和(2)必然在流场中的各个地方都适用。它们分别描述了可压缩流体和不可压缩流体的流体体积函数的输运规律。

运动界面追踪问题的控制方程一般由具体的运动介质物理场的数学模型(基本控制方程)和流体体积函数方程构成。对于介质面、自由面、间断面或者各种内部的运动界面, 可以相应

的、灵活的定义流体体积函数形式。基本控制方程和流体体积函数方程可以是耦合形式的。此外,若更细致的考虑运动界面对流场产生的应力效应项的贡献,相应的应力条件也应当加入到基本控制方程中去。

本文将讨论、分析和评估多种解决运动界面追踪和重构问题的高精度和高分辨率方法,这些方法都是建立在流体体积函数方程的基础上的。

1.2 运动界面重构问题的综述

由于界面是通过一系列时间步进行计算的,开始十分锐利的界面轮廓线在数值色散和耗散的影响下,有可能变得越来越模糊或者产生振荡。因而解决重构问题需要先重构运动界面轮廓。当前,许多文章提出了不同的界面重构方法。因此很难一一介绍和分析这些文章中的方法。本文将综合一些主流的和重要的方法。

一般说来,一个运动界面重构问题包含以下几个基本步骤。

1) 运动界面的初始位置

首先,为了更精确的追踪运动界面,流场中界面的初始条件应当是确定的,并且应尽可能精细地表达。

以 Zalesak 问题为例(如图 3 所示)。一个简单的方法可以通过定义在界面内的目标流体 $C = 1.0$, 而界面外的流体 $C = 0.0$ 。这样,所有的单元格都被指定为 $0 \leq C_{ij}^n \leq 1$ 。

2) 向量场 $V = (u, v)$ 的数值计算

在界面的初始条件和流场的边界条件给定后,在介质场的控制模型基础上,随时间推移将一步步地对界面进行数值计算。在此过程中不断及时地检查运动界面的情况。一旦需要时,即开始对界面进行重构。

3) 运动界面的重构

假设在 $t = t_n$ 时刻流体体积函数的单元分布为 $0 \leq C_{ij}^n \leq 1$ 。并发现界面光滑效应严重。需立刻开始界面的重构。

首先重构界面的斜率(如 Hirt 和 Nichols 的 FLOW_3D 方法^[1], Ashgriz 和 Poo 的 FLAIR 方法^[2], Noh 和 Woodward 的 SLIC 方法等等)或界面的法向量(如 Gueyffier^[7] 梯度向量差分方法和 Youngs 的 VOF^[8] 方法,等等)。需要考虑和比较所有目标单元周围的单元。通常可以采取有限差分方法^[7] 或一些有效的精细的重构方法,诸如代数^[2~4], 几何^[5~11], 或插值逼近等等。

在运动方向上进行计算并得到下一时刻单元内界面的情况 $0 \leq C_{ij}^{n+1} \leq 1$ 。这是运动界面追踪方法的关键。

2 特征型积分平均方法

一般说来,运动界面的捕捉方法应当是高精度数值方法,尤其是针对解决流体体积函数的对流程需要具有高分辨率的效果。一般在解方程(2)时它不需要对界面进行构造。运动界面的轮廓线可以由界面的几条等值线获得。这种方法在编程和实现中非常灵活。有时可以得到非常好的结果。

考虑一维例子

$$\partial C / \partial t + V(\partial C / \partial x) = 0 \quad (V > 0), \quad (3)$$

需要经过如下的步骤通过积分平均型格式构造方程的解

1) 考虑一个时空的控制元 $[x_{i-1/2}, x_{i+1/2}] \times [t_n, t_{n+1}]$, 并对方程(3)在此上积分

$$\int_{t_n}^{t_{n+1}} \int_{x_{i-0.5}}^{x_{i+0.5}} \frac{\partial C}{\partial t} + V \frac{\partial C}{\partial x} dx dt = 0 \quad (V > 0),$$

定义积分平均量

$$C_i^n = \frac{1}{\delta x} \int_{x_{i-0.5}}^{x_{i+0.5}} C^n(x) dx \quad (\delta x = x_{i+0.5} - x_{i-0.5}),$$

$$\langle VC_{i+0.5}^n \rangle = \frac{1}{\delta t} \int_{t_n}^{t_{n+1}} VC_{i+0.5}(t) dt \quad (\delta t = t_{n+1} - t_n).$$

因而原积分可写为

$$C_i^{n+1} = C_i^n - r(\langle VC_{i+0.5}^n \rangle - \langle VC_{i-0.5}^n \rangle) \quad (r = \frac{\delta t}{\delta x}). \quad (4)$$

2) 时间积分平均的离散。利用特征线方法, 可以得到

$$\langle C_{i+0.5}^n \rangle = \frac{1}{\delta t} \int_{t_n}^{t_{n+1}} C(x_{i+0.5}, t) dt = \frac{1}{\delta t} \int_n^{n+1} C \left(x_i + \frac{1}{2} \delta x - V(t - t_n), t_n \right) dt. \quad (5)$$

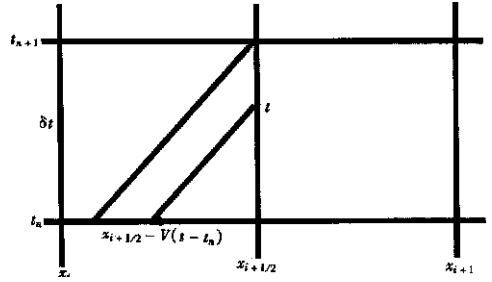


图1 利用特征线思想进行的积分变换

如果我们对于上述积分采用简单的梯形公式, 并且采用被积函数的一阶 Taylor 展开, 则有

$$\langle C_{i+0.5}^n \rangle = \frac{1}{\delta t} \left[C_i^n + \left(\frac{1}{2} \delta x - V \delta t \right) \frac{\partial C_i^n}{\partial x} \right] + \left[C_i^n + \left(\frac{1}{2} \delta x - 0 \right) \frac{\partial C_i^n}{\partial x} \right] / 2 \cdot \delta t = C_i^n + \frac{1}{2} (\delta x - V \delta t) \frac{\partial C_i^n}{\partial x} = C_i^n + \frac{1}{2} (1 - c) \frac{\partial C_i^n}{\partial x} \delta x = C_i^n + \frac{1}{2} (1 - c) \delta x S_i^n. \quad (6)$$

其中, c 是 Courant 数, S_i^n 是空间控制元上的斜率

$$c = V \delta t / \delta x, \quad S_i^n = \partial C_i^n / \partial x.$$

类似地有

$$\langle C_{i-0.5}^n \rangle = C_{i-1}^n + 0.5(1 - c) \Delta C_{i-1}^n = C_{i-1}^n + 0.5(1 - c) \delta x S_{i-1}^n. \quad (7)$$

3) 将(6)、(7)代入(4)式合成积分平均格式

$$C_i^{n+1} = C_i^n - c(C_i^n - C_{i-1}^n) + 0.5c\delta x(c - 1)(S_i^n - S_{i-1}^n). \quad (8)$$

对于一般的速度 V , 可将格式写为

$$C_i^{n+1} = C_i^n - r \max(0, V)(C_i^n - C_{i-1}^n) + 0.5r\delta x \max(0, V)(c - 1)(S_i^n - S_{i-1}^n) - r \min(0, V)(C_{i+1}^n - C_i^n) + 0.5r\delta x \min(0, V)(c + 1)(S_{i+1}^n - S_i^n). \quad (9)$$

如果取斜率 S_i^n 为

$$S_i^{LW, n} = (C_{i+1}^n - C_i^n) / \delta x. \quad (10)$$

就得到 Lax_Wendroff 格式。众所周知, Lax_Wendroff 不是保单调的, 在间断处会产生振荡。因此我们对斜率加以限制, 取

$$S_i^n = \phi(\theta_i^n) S_i^{LW, n}. \quad (11)$$

其中 $\phi(\theta^n) = \max(0, \min(1, 0, 2\theta^n), \min(2, 0, \theta^n))$ 是 Superbee 限制器。

对于二维和三维情形, 我们可以采用分数步法进行计算, 或者直接采用格林公式和有限体积法进行计算。

通过这种方法构造的格式具有相当高的精度和分辨率。我们也可以构造分片的二次或更高次的多项式来对未知函数进行逼近。(例如, PPM 方法或 ENO 和 Weighted ENO 方法)。我们

利用上述的特征型积分平均有限体积法加上适时重构,对旋转速度场和剪切速度场的运动界面问题进行了计算,得到了很好的效果。

3 计算结果和算法比较

本节给出了对流方程(2)的一些数值试验,模拟了三个模型速度场问题:常数速度场,旋转速度场和剪切速度场。并且与 FCT_VOF, FLAIR_VOF, Youngs 方法和 Ubbink 和 Issa 的 CICSAM 等方法进行了相应的比较。

3.1 常数速度场(平移)

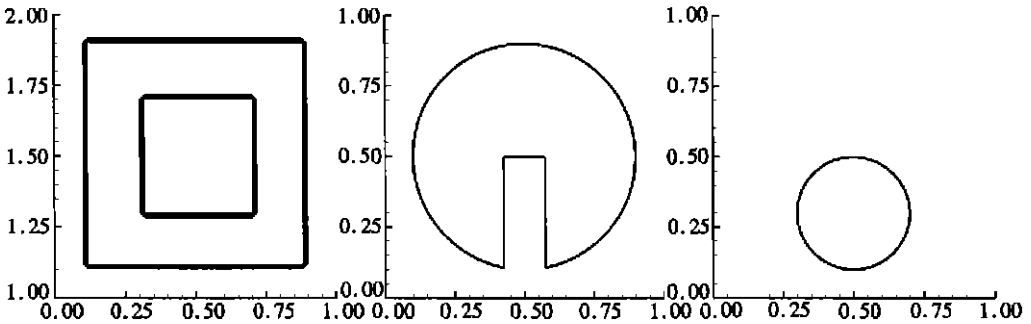


图2 平移速度场初值

图3 旋转速度场初值

图4 剪切速度场初值

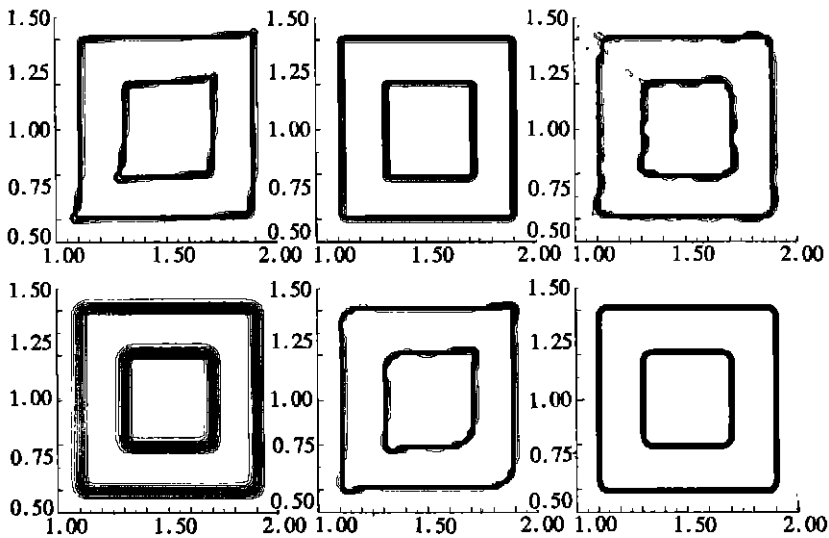


图5 平移速度场数值结果

考虑一个常数速度场 $(u, v) = (1, 0)$ 或 $(u, v) = (2, -1)$ 。我们取初值为如图2所示的同心方框。整个计算区域为 $[0, 2] \times [0, 2]$ 。体积函数在两方框之间的区域取为 $C = 1.0$, 在其他区域取 $C = 0.0$ 。网格大小取 200×200 。时间步长取为 $\delta t = 0.001$ 。图5给出了速度场 $(u, v) = (2, -1)$, 时间 $t = 0.5$ 时上述六种方法的计算结果。图中显示的是从 0.0625 到 0.9375 的 15 条等值线。很明显,几乎所有的方法都能给出较好的结果。但是 FLAIR_VOF 和 Hirt_VOF 方法不如其他的那么好。而积分平均方法, FCT_VOF 方法, Youngs_VOF 方法和 CICSAM 给出了比较好的结果。Youngs_VOF 方法在角点处出现光滑化,这是由于求法向量的算法引起的。对于相对光滑的界面, Youngs_VOF 方法就比 CICSAM 方法要好一些。各种方法计算

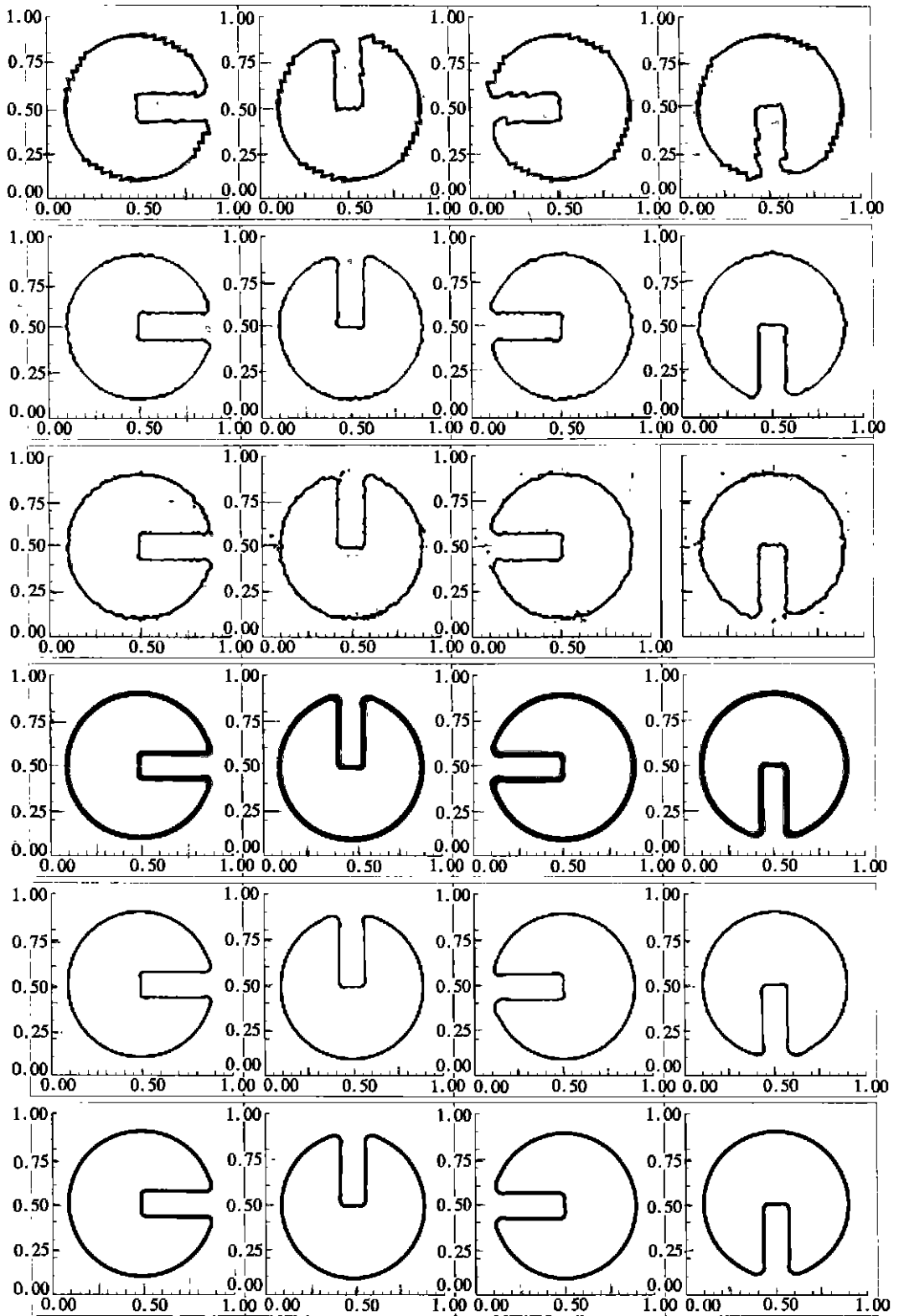


图6 旋转速度场数值结果(从左到右时间 t 依次为 0.5, 1.0, 1.5, 2.0)

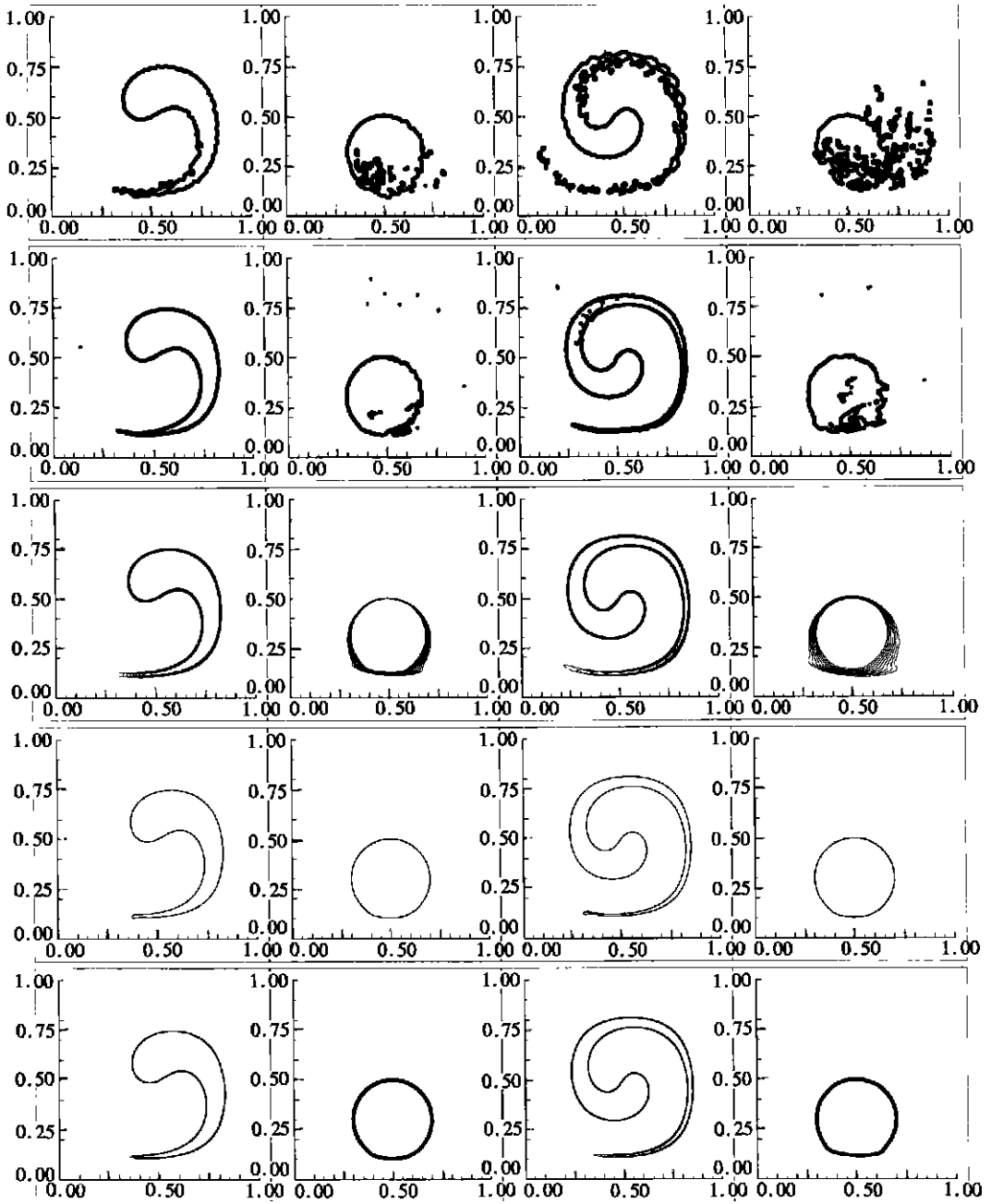


图7 剪切速度场数值结果

(从左到右依次为 $t = 1.0$ 及其反剪切, $t = 2.0$ 及其反剪切)

结果如图5所示。

3.2 旋转速度场(Zalesak 问题)

考虑二维旋转速度场

$$u(x, y) = -\pi(y - 0.5), \quad v(x, y) = \pi(x - 0.5) \quad (12)$$

这是一个检验上述方法效果的很好的算例。取计算区域 $[0, 1] \times [0, 1]$ 。取初值(Zalesak 界面, 如图3所示)为有缺口的圆周。要在尖锐的角点处保持高分辨率是很困难的。我们取网格大

小为 200×200 , $\delta t = 0.001$ 。在速度场(12)的作用下,初值图分别旋转 $\pi/2, \pi, 3\pi/2$ 和 2π 。结果如图 6 所示。可以看出, Hirt 和 Nichols 的 VOF 方法相当粗糙。而积分平均方法, Youngs 的 VOF 方法, FCT_VOF 方法和 CICSAM 方法得到了非常好的结果, 而以 CICSAM 和积分平均方法的结果最好。

3.3 剪切速度场

平移和旋转流场都不会使运动界面的形状发生变化。而在一个真实的流场问题中, 会包含如拉伸、剪切、碰撞、破碎等复杂的物理现象。为了考察算法对各种现实流场的模拟效果, 考虑一个剪切流场

$$\begin{cases} u(x, y) = \pi \cos(\pi(x - x_0)) \sin(\pi(y - y_0)), \\ v(x, y) = -\pi \sin(\pi(x - x_0)) \cos(\pi(y - y_0)). \end{cases} \quad (13)$$

取 $(x_0, y_0) = (0.5, 0.5)$, 计算区域取 $[0, 1] \times [0, 1]$, 初值界面为圆心在 $(0.5, 0.3)$ 、半径为 0.2 的圆周(如图 4)。用上述除 Hirt_VOF 之外的五种方法, 分别计算到 $t = 1.0$ 和 $t = 2.0$ 后。再以此结果为初值, 将速度反号, 分别反剪切到 $t = 1.0$ 和 $t = 2.0$, 结果如图 7 所示。可以见到, FCT_VOF 和 FLAIR_VOF 的结果不好。而 Youngs_VOF、Ubbink 的 CICSAM 方法和积分平均方法得到了很好的结果。其中, Ubbink 的 CICSAM 方法得到的结果最好。甚至对于其反转后的结果, 界面的锐利情况和初始值几乎一样。

4 结 论

在这篇文章中, 我们提出了一种高精度和高分辨率方法——特征型积分平均有限体积法。并给出了有关运动界面重构方法的评估和数值模拟。通过对三种不同模型的数值模拟, 比较了各种数值方法的精度。我们的计算结果显示, Youngs_VOF, Ubbink 的 CICSAM 和积分平均方法是最为有效的, 并且具有高精度和高分辨率。

附录 A: 界面重构型方法

界面重构型方法通过对运动界面的重构来数值模拟(1)和(2)。这类方法具有很强的物理背景。一般说来, 它们是守恒的, 但也比较复杂。

A.1 施主_受主方法^[1]

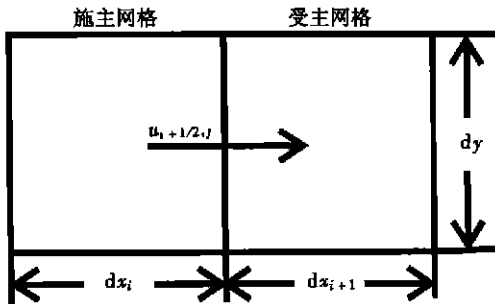


图 A1 施主_受主网格

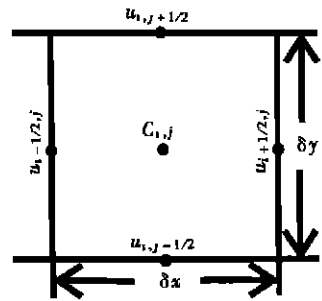


图 A2 计算网格

Hirt 和 Nichols 于 1981 年提出了一种重构的方法。在这种方法中, 界面是通过水平和垂直的直线来近似得到的。如果在一个网格内界面的倾斜度大于 $\pi/4$, 则该网格内的运动界面被看作是水平的, 反之则被看作是垂直的。首先, 通过在网格边界的速度确定俩个相邻网格之间的关系。处在上游的网格被称为施主(Donor)网格, 处在下游的网格被称为受主(Acceptor)网格(如图 A1)。计算的网格(如图 A2)所示。

通过网格边界上的对流求解方程(2)。取控制体

$$\Omega = (x_{i-1/2}, x_{i+1/2}) \times (y_{j-1/2}, y_{j+1/2}) \times (t_n, t_{n+1}),$$

在其上对方程(2)积分,我们得到

$$(C^{n+1} - C^n) \delta x \delta y + \Delta(Cu) \delta t \delta y + \Delta(Cv) \delta t \delta x = 0, \tag{A1}$$

在施主_受主方法中,我们定义

$$\Delta(Cu) \delta t = \min \{ C_{AD} |V_x| + C_F, C_D \delta x_D \},$$

其中, $C_F = \max \{ (1.0 - C_{AD}) |V_x| - (1.0 - C_D) \delta x_D, 0 \}$, $|V_x| = |u \cdot \delta t|$. 下标 A, D 分别表示受主单元和施主单元. C_A, C_D 分别表示受主和主单元上的流体体积函数. 双下标 AD = A 或 D, 取决于和流动方向相关的运动界面的方向. 若施主界面是垂直的, 则采用受主单元上的流体体积函数(也就是界面的方向基本和流体速度的法线方向一致), 反之, 则采用施主单元上的流体体积函数. $\Delta(Cv) \delta t$ 的构造方法类似.

通过这种方法, 采用了施主和受主两种单元上的流体体积函数, 提高了分辨率. 然而, 由于只通过水平或垂直线来对界面进行近似, 因而这种方法还是比较粗糙的.

A.2 FLAIR 方法^[2]

Ashgriz 和 Poo 的 FLAIR(Flux Line_segment model for Advection and Interface Reconstruction) 方法采用了更为精确的斜线段来近似格子内的界面. 对共有一条网格边界的两个相邻网格(如图 A3 所示), 构造一条直线段作为跨过该网格边界的界面近似. 然后可以计算一个时间步内流过该网格边界的流体体积量. 这可以作为修改流体体积函数的数值流通量. 由于界面构造涉及到两个网格, 需要分成多种情况分别进行计算. 利用网格边界的流体速度, 确定施主单元和受主单元. 其流体体积函数分别为 C_D, C_A 表示. 然后可以分为以下几种计算:

- 1) 施主网格是满网格, $C_D = 1.0$; 2) 施主网格是空网格, $C_D = 0.0$;
- 3) 施主网格和受主网格既不是满网格也不是空网格, $0 < C_D < 1.0, 0 < C_A < 1.0$
- 4) 施主网格既不是空网格也不是满网格, 受主网格为空或者为满网格, 即 $0 < C_D < 1.0, C_A = 0.0$ 或 $C_A = 1.0$

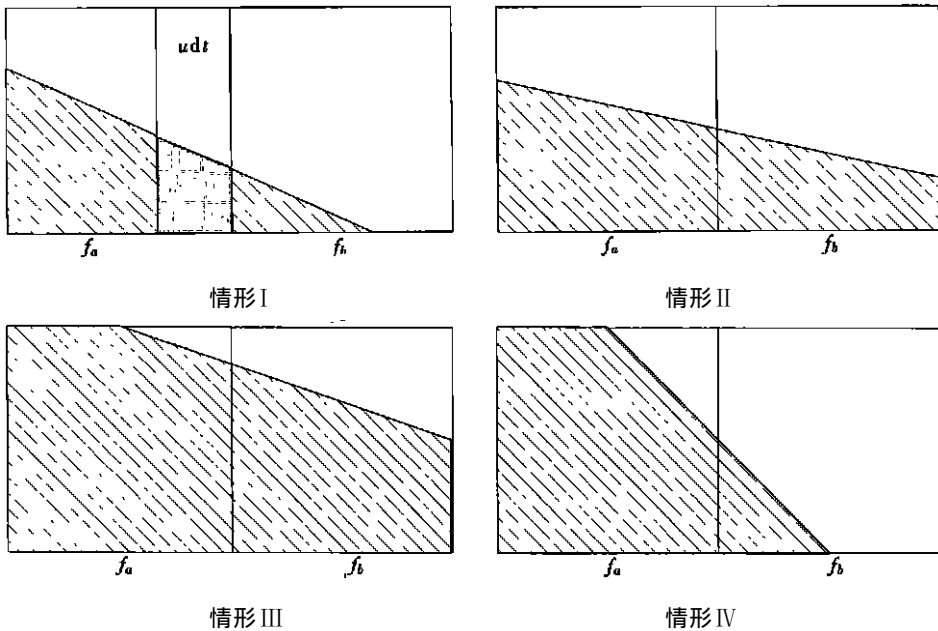


图 A3 FLAIR 界面重构的四种情形

前两种情况比较简单. 第三种情况又可以被分为 16 种情形. 经过简单的转换可以归结为 4 种, 即 $f_a \geq f_b$, 且目标流体在界面以下(如图 A3). 构造一条直线穿过两个网格, 其下阴影区域是目标流体. 由两个网格的流体体积函数来确定这条直线的斜率和位置. 从而计算出在一个时间步内通过该网格边界的流体流量. 比如在第一种情况下(情形 I), 假设左边是施主网格, 则一个时间步内被运输到相邻网格的流体体积量就是宽度为 $u \delta t$ 斜线阴影区域. 假设右边是施主网格, 则应取右边网格内靠近网格边界的宽度为 $u \delta t$ 的区域面积.

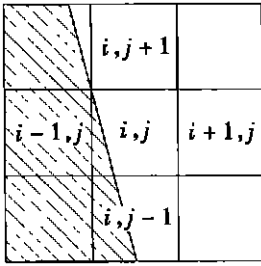


图 A4 目标格子与其临格

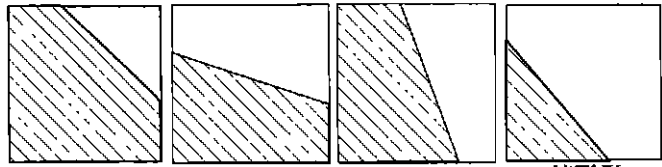


图 A5 目标格子内运动界面的四种情况

第四种情况更加复杂 如图 A4 所示 施主网格 (i, j) 为半网格, 受主网格 $(i+1, j)$ 是空网格 我们可以用网格 $(i, j+1)$ 和 $(i, j-1)$ 的另一个方向的两个相邻网格网格分别按 3) 的方法构造界面 首先计算两相邻网格的界面斜率, 再利用所得的斜率平均值作为网格 (i, j) 内界面的近似斜率 然后按类似于 3) 的方法构造直线作为此网格内的界面近似 然后求出在一个时间步内流过网格边界流入受主网格的流体体积量 这种情况下的其他情况同样可以归结为如图 A5 所示的 4 种情形 如果受主网格为空网格, 则取靠右边的宽度为 $u\delta t$ 的区域面积作为流通量 反之, 则取左边的宽度为 $u\delta t$ 的区域面积作为流通量

同样的, 实际计算中采用方向分裂算法 首先根据 x 方向的流体速度计算所有 x 方向的数值流通量, 对流体体积函数作修改 由此得到的所有网格上的流体体积函数的中间值用于对 y 方向做同样的计算 以此得到下一时刻的流体体积函数 我们的试验表明, 如果同时对 x 和 y 方向的流通量进行计算, 则会产生质量不守恒

A.3 Youngs 方法^[8]

与 FLAIR_VOF 方法不同, Youngs 方法是在单个网格内用直线段近似界面 先计算在目标网格 (i, j) 内界面的法向 $\mathbf{n} = (n_{i,j}^x, n_{i,j}^y)$

$$\begin{cases} n_{i,j}^x = (C_{i+1,j+1} + 2C_{i+1,j} + C_{i-1,j-1} - C_{i-1,j+1} - 2C_{i-1,j} - C_{i-1,j-1})/\delta x, \\ n_{i,j}^y = (C_{i+1,j+1} + 2C_{i,j+1} + C_{i-1,j+1} - C_{i+1,j-1} - 2C_{i,j-1} - C_{i-1,j-1})/\delta y, \end{cases} \quad (A2)$$

根据法向, 可以确定运动界面与 x 轴的夹角 β

$$\beta = \arctan\left(-\frac{n^y}{n^x}\right), \quad \alpha = \arctan\left((\delta x/\delta y)\tan\beta\right) \quad (\pi/2 \leq \alpha \leq \pi), \quad (A3)$$

与 FLAIR 方法类似, 也有 16 种情形 通过简单的对称和反转, 可以化为如图 A5 所示的四种情形 利用此夹角和网格内流体体积函数, 我们可以确定属于哪一类型 然后计算直线的斜率和位置, 构造该网格内的界面 并计算在一个时间步内流过四周到相邻网格的流体体积量 以修改本网格和四周相邻网格的流体体积函数值

用这种方法得到的结果相当精确 但是由于 x 和 y 方向的流通量是同时计算的, 因而质量的守恒方面遭到了一定的破坏

A.4 Ubbink 和 Issa 的 CICSAM 方法^[11]

以上所有求解 VOF 方程的方法, 在二维和三维的问题中大多都是采用方向分裂或算子分裂方法 近来, Ubbink 和 Issa 提出的 CICSAM (Compressiv Interface Capturing Scheme for Arbitrary Mesh) 方法, 通过半隐式处理 VOF 方程, 使得该方法不用通过算子分裂就可以推广到多维问题 另一个显著的地方在于此方法在无结构网格仍然是有效的 在 CICSAM 方法中采用了 NVD (Normalized Variable Diagram) 的概念, 以及 Hirt 和 Nichols 的主要思想, 即是采用耗散格式 (如迎风格式, 以得到光滑的界面) 还是采用色散格式 (如逆风格式, 以得到锐利的界面) 是通过自由面方向决定的 并且在色散格式和耗散格式之间引入了一个切换参数以提高色散和耗散格式的精度 该方法在获得锐利的界面的同时又保证了体积函数的有界性

在交错网格上 (如图 A2), 除了速度是定义在网格边界上以外, 方程 (2) 中的其他变量都是定义在网格的中心点上的 采用 Crank_Nicolson 离散

$$(C_p^{n+\delta t} - C_p^n) V_p = - \sum_{f=1}^n \frac{1}{2} ((C_f E_f)^{t+} + (C_f F_f)^{t+\delta t}) \delta t, \quad (A4)$$

其中, p 代表网格 (控制体) 的中心, f 表示网格的边, n 表示网格的边界数目, V_p 表示控制的体积, F_f 表示通过

边界 f 向外的流通量。网格边界上的流体体积函数 C_f 可以通过相邻的施主网格和受主网格的流体体积函数的平均表示

$$C_f = (1 - \beta_f) C_{D+} + \beta_f C_A, \quad (A5)$$

由于时间步足够小,可认为 $\beta_f^{t+\delta t} = \beta_f^t$ 则(A4)可以写成二阶精度形式

$$C_p^{t+\delta t} = C_p^t - \frac{\delta t}{V_p} \sum_{f=1}^n C_f^* F_f, \quad (A6)$$

$$\text{其中 } C_f^* = (1 - \beta_f) \frac{C_D^t + C_D^{t+\delta t}}{2} + \beta_f \frac{C_A^t + C_A^{t+\delta t}}{2}, \quad (A7)$$

网格边界上 β_f 的确定用NVD方法^[11]。

- 1) 计算归一化变量 C_D, C_U 为施主网格的上游网格

$$C_D = (C - C_U) / (C_A - C_U) \quad (A8)$$

- 2) 计算CBC(Convection Boundedness Criteria)的上界 C_{fCBC}

$$C_{fCBC} = \begin{cases} \min \left\{ \frac{C_D}{c}, 1.0 \right\} & (\text{如果 } 0 \leq C_D \leq 1), \\ C_D & (\text{其他}), \end{cases} \quad (A9)$$

这是 c 是施主单元的Courant数,定义为

$$c = \sum_j |c_f^{\text{out}}|,$$

这是 c_f^{out} 是施主单元的每个流出面的Courant数。

- 3) 计算归一化变量 C_{fUQ} (Ultimate_Quickest格式)

$$C_{fUQ} = \begin{cases} \min \left\{ \frac{8cC_D + (1-c)(6C_D + 3)}{8}, C_{fCBC} \right\} & (\text{如果 } 0 \leq C_D \leq 1), \\ C_D & (\text{其他}), \end{cases} \quad (A10)$$

该格式具有较强的耗散性。单独使用会抹平界面。但与其他格式配合使用可以消除振荡。

- 4) 计算加权平均值

$$\begin{cases} C_f = \gamma_f C_{fCBC} + (1 - \gamma_f) C_{fUQ}, & \gamma_f = \min \{ k_v 0.5 (\cos(2\theta_f) + 1), 1 \} \\ \theta_f = \arccos |(\cdot \cdot C)_D \cdot \mathbf{d}_f| / |(\cdot \cdot C)_D| \cdot |d_f|. \end{cases} \quad (A11)$$

其中 \mathbf{d}_f 是连接施主和受主网格中心的矢量。如果网格是矩形,则该矢量与边上的法向同向。 $k_v \geq 0$ 是控制两种格式权值的参量。Ubbink和Issa方法建议 $k_v = 1$ 。

对于不同的迎风和逆风的切换准则,会导致结果相差很大。CICSAM方法采用在Hyper_C格式和Ultimate_Quickest格式间一种非常光滑的切换方式。这种切换也是基于界面与边的交角的。如果界面与边平行,则趋向于采用耗性的格式来光滑界面。反之,界面与边垂直时,则趋向于采用色散性的格式,以获得锐利的界面。

- 5) 根据归一化变量确定 β_f

$$\beta_f = (C_f - C_D) / (1 - C_D). \quad (A12)$$

通过以上的步骤有可能产生非物理值。或者说,取值不符合有界性条件 $0 \leq C \leq 1$ 。这时需要对流体体积函数进行有界性修正。Hint和Nichols直接采用强制性条件使得体积函数分别等于0或1,但这会导致流体体积函数的不守恒从而影响动量方程而使得计算出的流场不真实。Ubbink和Issa和CICSAM针对上述问题采用了一种对格子边上的体积通量进行修正的方法^[11]。

[参 考 文 献]

- [1] Hirt C W, Nichols B D. Volume of Fluid(VOF) method for the dynamics of free boundaries[J]. J Comput Phys, 1981, **39**: 201—225.
- [2] Ashgriz N, Poo J Y. Flair: Flux line_segment model for Advection and interface reconstruction[J]. J Comput Phys, 1991, **93**(2): 449—468.
- [3] Rudman M. Volume_tracking methods for interfacial flow calculations[J]. Int J Numer Meth Fluids, 1997, **24**(7): 671—691.

- [4] Rudman M. A volume_tracking method for incompressible multifluid flows with large density variations[J]. *Int J Numer Meth Fluids*, 1998, **28**(2): 357—378.
- [5] Kim S O, No H C. Second_order model for free surface convection and interface reconstructions[J]. *Int J Numer Meth Fluids*, 1998, **26**(1): 79—100.
- [6] Rider W J, Kothe D B. Reconstructing volume tracking[J]. *J Comput Phys*, 1998, **141**(2): 112—152.
- [7] Gueyffier D, Li Jie. Volume_of_fluid interface tracking with smoothed surface stress methods for 3D flows[J]. *J Comput Phys*, 1999, **152**(2): 423—456.
- [8] Youngs D L. Time_dependent multi_material flow with large fluid distortion[A]. In: Morton K W, Baines M J Eds. *Numerical Methods for Fluid Dynamics*. Academic[C]. New York: Academic, 1982.
- [9] Zalesak S T. Fully multidimensional flux_corrected transport algorithms for fluids[J]. *J Comput Phys*, 1979, **31**: 335—362.
- [10] Vincent S, Caltagirone J P. Efficient solving method for unsteady incompressible interfacial flow problems[J]. *Int J Numer Meth Fluids*, 1999, **30**(6): 795—811.
- [11] Ubbink O, Issa R I. A method for capturing sharp fluid interfaces on arbitrary meshes[J]. *J Comput Phys*, 1999, **153**(1): 26—50.
- [12] Osher S, Sethian J A. Fronts propagating with curvature_dependent speed: algorithms based on Hamilton_Jacobi formulations[J]. *J Comput Phys*, 1988, **79**: 12.
- [13] Sethian J A. *Theory, Algorithms, and Applications of Level Set Methods for Propagating Interface* [M]. Acta Numerica, Cambridge UK: Cambridge Univ Press, 1995.
- [14] Sethian J A. *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Material Sciences* [M]. Cambridge UK: Cambridge Univ Press, 1996.
- [15] Sussman M, Smereka P, Osher S. A level set approach for computing solutions to incompressible two_phase flow[J]. *J Comput Phys*, 1994, **114**(1): 146—149.
- [16] Chen S, Merriman B, Osher S, et al. A simple level set method for solving stefan problems[J]. *J Comput Phys*, 1997, **135**(1): 8—29.

Tracking and Reconstruction Methods for Moving Interfaces

LIU Ru_xun^{1,2}, LIU Xiao_ping², ZHANG Lei^{1,3}, WANG Zhi_feng¹

(1. Department of Mathematics, University of Science and Technology of China,
Hefei 230026, P. R. China;

2. State Key Laboratory of Fire Science of University of Science
and Technology of China, Hefei 230027, P. R. China;

3. School of Basic Science, Chinese University of Mining and Technology,
Beijing 100083, P. R. China)

Abstract: A simple, efficient and accurate high resolution method to tracking moving interfaces—the characteristic integral—averaging finite volume method on unstructured meshes is proposed. And some numerical tests and evaluation of six main efficient methods for interface reconstruction are made. Through strict numerical simulation, their characters, advantages and shortcomings are compared, analyzed and commended in particular.

Key words: moving_interface; interface_reconstruction; fluid volume function; integral_averaging; finite volume method