

文章编号: 1000-0887(2004) 10-0991-06

# 一种改进的 Level\_Set 函数 重新初始化方法\*

王志亮, 周哲玮

(上海大学, 上海市应用数学和力学研究所, 上海 200072)

(我刊编委周哲玮来稿)

**摘要:** 改进了 level\_set 界面捕捉方法中的 level\_set 函数重新初始化方法, 通过锁定界面位置来实现 level\_set 函数重新初始化过程, 较好地保持了在重新初始化过程中的界面位置的准确性, 保障了流体体积的相对稳定. 对剪切运动使流体界面变形的算例表明方法对 level\_set 函数重新初始化过程中的体积保持比较有效, 而重新初始化过程总的机器用时并没有显著增加.

**关键词:** level\_set 方法; 体积守恒; 重新初始化

**中图分类号:** O358      **文献标识码:** A

## 引 言

Level\_set 方法(Osher & Sethian (1988)<sup>[1]</sup>, Mulder, Osher & Sethian (1992)<sup>[2]</sup>)是一种用来描述界面位置的方法, 流动空间被等梯度距离函数所覆盖, 而距离函数在流体界面上取值为零. 距离函数随着流场结构变化而迁移, 零等值面也随着界面位置的迁移而变化. level\_set 方法是一种很好的界面捕捉方法, 在计算过程中不管流场如何改变, level\_set 函数始终是光滑的; 对诸如破碎、合并、膨胀之类的界面变形或拓扑改变可以非常自然地进行处理. 但 level\_set 方法不是一种体积守恒方法, 在一般的情况下求解的过程总是伴随着体积的损失, 由于体积损失意味着界面计算结果对正确位置的偏移, 从而造成了流场的失真. 很多研究者为了解决体积损失问题作出了不懈的努力.

Fatemi, *et al.* (1995)<sup>[3]</sup>, Sussman & Smereka (1997)<sup>[4]</sup>, Sussman, *et al.* (1999)<sup>[5]</sup> 为了保持体积在流体单元中守恒, 发展了一种类似于 VOF (volume of fluid method) 体积函数的方法, 采用 9 节点模板格式计算流体单元流体体积, 在对 level\_set 函数进行重新初始化的过程中, 对计算过程中流体体积的修正量, 通过最初的流体体积来计算, 然后通过这个修正量来改变距离函数 (level\_set 函数). 与他们的方法不同, Chang, *et al.* (1996)<sup>[6]</sup> 注意到界面数值扩散与当地的曲率有

\* 收稿日期: 2003\_10\_12; 修订日期: 2004\_05\_31

基金项目: 国家自然科学基金资助项目(19772026); 上海市科委科研基金资助项目(1998\_10); 上海市重点学科建设资助项目; 上海市属博士学科点建设基金项目

作者简介: 王志亮(1974—), 男, 湖北人, 博士(Tel: + 86\_21\_56772253; E\_mail: wangzl\_one@163.com);

周哲玮(1950—), 男, 湖北人, 博导, 教授, 博士(联系人, Tel: + 86\_21\_56334092;

E\_mail: zhwzhou@yc.shu.edu.cn)•

关, 以此为依据而设计了一个发展方程, 通过方程的时间发展趋向于稳态解的特性来保持流体体积的守恒. Chang, *et al.* 的方程具有如下形式:

$$\frac{\partial \phi}{\partial t} + (A_0 - A(t))(-P + \kappa) |\nabla \phi| = 0,$$

$$\phi(\mathbf{x}, 0) = \phi_0(\mathbf{x}),$$

其中,  $\phi_0(\mathbf{x})$  是初始的 level\_set 函数,  $A_0$  是初始的总体积,  $A(t)$  是计算过程中的总体积,  $P$  是一个正的经验常数,  $\kappa$  为当地的曲率.

无论是 Fatemi, *et al.* 或是 Chang, *et al.* 的方法, 都必须考虑到有关于流体体积的计算问题.

本文提出了一种新的 level\_set 函数的重新初始化方法. 文章的安排是这样的: 第 1 节将对造成体积损失的可能原因进行分析; 第 2 节将提出解决方案; 第 3 节将给出一个算例; 第 4 节将对这一问题进行总结.

### 1 Level\_set 函数的重新初始化算法

Level\_set 函数的重新初始化过程通常是求解一个 Hamilton\_Jacobi 型系统. 在二维情形有如下形式:

$$\phi_t + S(\phi_0) \left[ \sqrt{\phi_x^2 + \phi_y^2} - 1 \right] = 0,$$

$$\phi(x, y; 0) = \phi_0(x, y),$$

其中  $\phi_0$  是初始 level\_set 函数,  $S(\phi_0)$  是符号函数. 随着时间发展, 方程最终收敛到一种定常的状态, 即  $|\nabla \phi| = 1$ . 对这样的一个系统, 一般采用单调 Godunov 格式 (Chen, *et al.* (1997)<sup>[7]</sup>) 或者修正的 Godunov 格式 (Fedkiw, *et al.* (1999)<sup>[8]</sup>) 来进行求解. 修正的 Godunov 格式是在单调 Godunov 格式的基础上对时间采用多步法, 对空间导数采用 ENO 格式或者 WENO 格式的高精度算法. 文中所采用的是时间单步, 空间导数采用 5 阶精度 WENO5 格式. 算法可以描述如下:

$$\phi_{i,j}^{n+1} = \phi_{i,j}^n - \Delta t S_\epsilon(\phi_{0,i,j}) G(\phi_{i,j}^n),$$

其中

$$S_\epsilon(\phi_{0,i,j}) = \frac{\phi_{0,i,j}}{\sqrt{\phi_{0,i,j}^2 + \Delta x^2}},$$

$$G(\phi_{i,j}^n) = \begin{cases} \sqrt{\max((a^+)^2, (b^-)^2) + \max((c^+)^2, (d^-)^2)}, & \phi_{0,i,j} > 0, \\ \sqrt{\max((a^-)^2, (b^+)^2) + \max((c^-)^2, (d^+)^2)}, & \phi_{0,i,j} < 0, \\ 0 & (\text{其他}), \end{cases}$$

$z^+ = \max(z, 0)$ ,  $z^- = \min(z, 0)$ , 其中  $a, b, c, d$  为空间导数(空间导数的 WENO5 格式可以参见 Jiang, *et al.* 的[9]).

这种算法的构造的特点是迎风差分(见图 1)及单调格式. 应用单调 Godunov 格式来跟踪界面具有很好的收敛性和收敛速度, 对二维的界面问题, 如果要达到 0.01 h 的精度通常只需

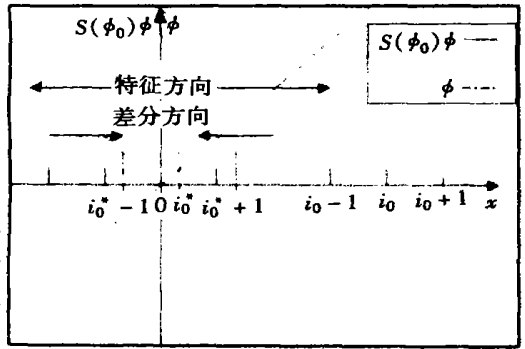


图 1 level\_set 方法迎风格式思想示意图

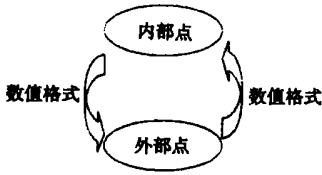


图2 节点数据的相互依赖

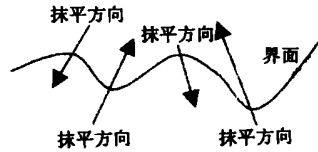


图3 数值界面抹平方向示意图

要3、4个时间步。但由于算法本身不守恒，level\_set 函数的重新初始化会造成体积损失。这种损失可能在一次计算中不会造成太大的影响，但随着流场在时间上的发展，在经过较长的一段时间之后，这种损失就相当可观，在这样的情况下，计算就会失去意义。通过分析和计算发现，造成偏差的主要原因可能有两个：其一、数据之间存在互动，这是数值算法的一个特点。由于算法是通过梯度将流场中的节点数据相互联系起来，数据在被其他数据改变的同时也改变其他的数据（见图2）。在这种情况下，如果数值计算结果的改变具有某种倾向性的话，那么相应的界面位置就会失真，从而造成体积的损失。其二、界面信息的不对称，这是 level\_set 函数的数值特性造成的。由于通常情况下界面位置是弯曲的，在计算的过程中可以注意到，体积的损失方向总是指向凹的方向（见图3），这是可以想象的。由于界面的凸面数据在数量上相对凹面数据具有优势，因此在计算过程中凸面数据所装载的信息更多地体现在计算当中，于是相应的界面就会被抹平，这是一种存在于界面的数值扩散现象。Chang, et al. (1996) 注意到的界面数值扩散与当地曲率的相关性也存在类似的机理，在数值计算过程中这类现象非常常见。所以在这两个因素间的相互作用下，界面位置在计算过程中发生了偏移和失真。

## 2 数值方案设计

针对以上造成 level\_set 函数在重新初始化过程中的体积损失原因的分析，就可以设计自己的数值方法。

为了阻止数值计算过程中界面漂移的发生，本文首先用插值方法将界面与网格线的交点固定，然后再通过 Godunov 格式进行 level\_set 函数的求解，生成 level\_set 函数空间。同时，采用单边生成距离函数然后再生成另外一边的方法，也可以加快 level\_set 函数生成的速度。

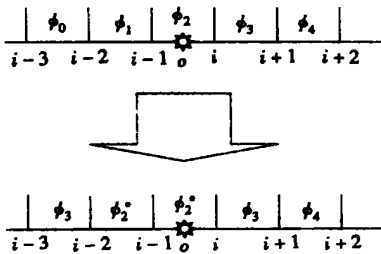


图4 空间导数的确定

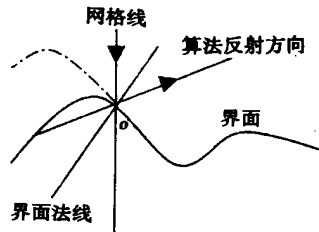


图5 函数在界面的反射

对于 level\_set 函数的空间导数，则采用在界面点两侧对称的策略（见图4）。在图4中， $o$  为 level\_set 函数零等值面与网格的交点（记为  $x_o$ ），即插值界面点， $\phi_i$  为对应网格点  $x$ （例如对 WENO5 格式  $x = (i - 2) \sim (i + 3)$  或  $x = (i - 3) \sim (i + 2)$ ）的空间导数。图中我们是在生成右边的 level\_set 函数，因此，离开  $o$  点最近的右侧空间导数值  $\phi_2$  由右侧界面附近点同界面的导数  $\phi_2^*$  代替，而其它左侧的空间导数则在界面点两侧呈对称分布，所以图4中左侧空间导

数值  $\phi_0, \phi_1$  被  $\phi_3, \phi_2^*$  分别替代。那么若将通常的方法表示为  $\phi_x = \text{WENO5}(\phi_0, \phi_1, \phi_2, \phi_3, \phi_4)$ , 则现在将表达式在界面改写为  $\phi_x = \text{WENO5}(\phi_3, \phi_2^*, \phi_2^*, \phi_3, \phi_4)$ , 对其它的格式操作也是类似的。采用这种方法, 对于一维的问题算法的作用和意义是明显的, 对于二维和高维的问题, 这样做的意义在于, 使离开界面网格点的  $\text{level\_set}$  函数值以最短的路径(梯度方向)获得(见图 5), 从而加快算法的收敛速度。同时我们也发现, 由于算法对界面的位置不产生移动效应, 所以采用低阶和高阶方法的效果区别不大, 因而采用低阶方法也不失为一种好的选择。

### 3 算 例

刚体平移或旋转运动不会使运动界面发生变形, 而剪切流场运动对界面的作用最为明显, 因此为了考察算法在剪切流动问题中的有效性, 将计算一个经典剪切流场作用下的二维流体输运问题, 其速度场为:

$$u(x, y) = \pi \cos(\pi(x - x_0)) \sin(\pi(y - y_0)),$$

$$v(x, y) = -\pi \sin(\pi(x - x_0)) \cos(\pi(y - y_0)),$$

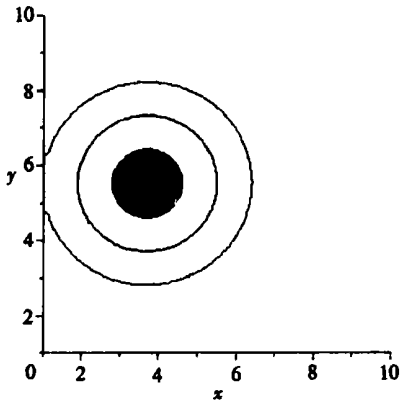


图 6 流场初始状态

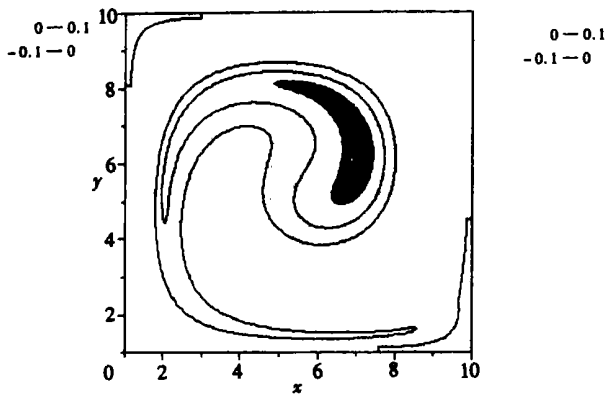


图 7 平均积分方法(  $t = 1.0 \text{ s}$  )

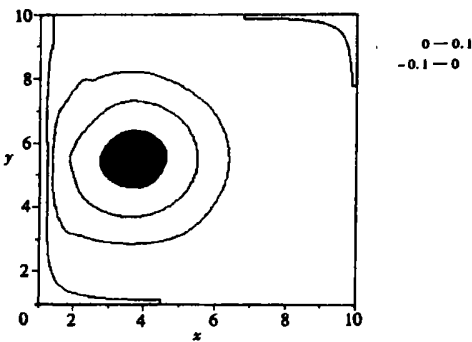


图 8 平均积分方法(  $t = -1.0 \text{ s}$  )

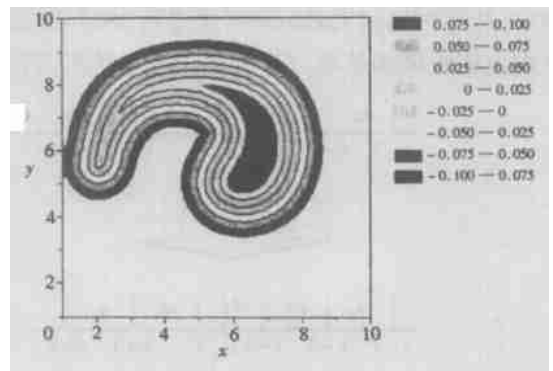


图 9 WENO5(  $t = 1.0 \text{ s}$  )

计算区域以及初始状态见图 6。将以此为初值计算到时间为 1.0 s, 然后再以 1.0 s 时的状态为初始条件进行逆向计算(记为  $t = -1.0 \text{ s}$ ), 考察它的还原效果。对于流动的计算采用积分平均格式, 这种格式对这类问题的计算效果比较好。采用的网格是  $128 \times 128$ , 时间步长  $\Delta t = \Delta h/10$ ,  $\Delta h$  为网格宽度。图 7 和图 8 分别为直接采用积分平均格式得到的时间为 1.0 s 和反算得还原结果, 这也将作为参考。

图9和图10分别采用积分平均TVD格式<sup>[10]</sup>计算流动场,传统的空间导数采用WENO5的修正的Godunov格式。图11和图12是本文算法所得,其中插值采用代数插值方法。

通过比较看到,相对而言,算法在波前的捕捉和体积保持上是有优势的。而且对于剪切流动通过类似方法是可以改善原来算法的计算效果的。

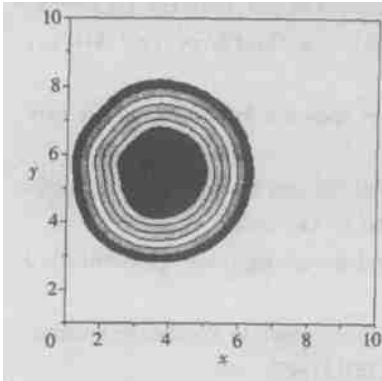


图10 WENO5( $t = -1.0$  s)

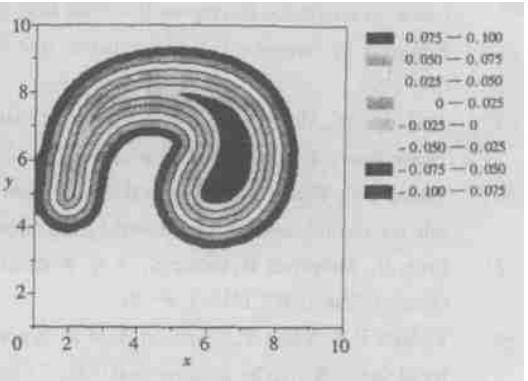


图11 本文的方法( $t = 1.0$  s)

从图9和图11比较可以看出本文的算法中流动的前端更加深入,虽然没有达到图7中的位置,但已经相当的吻合了,而且图11中流体的轮廓比图9中大,这一点不是非常明显,但在图10和图12中可以看得比较清楚。比较图8、图10和图12还原后的流体,可以发现计算在流速较大的外围有较好的表现。在图8中靠近壁面的外围体积损失比较明显;图10中流体的轮廓比较平滑,但体积收缩;图12中还原后的流体体积保持较好,但在外围靠近壁面的位置隐约存在一个突起。经过分析,发现造成突起的主要原因是流动前端的界面插值精度不高,而由插值点选取的不对称使得界面计算位置向外偏离了实际位置。

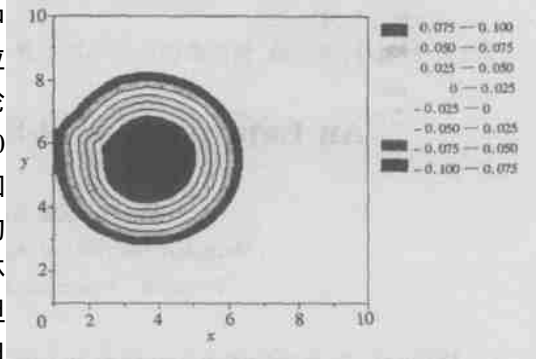


图12 本文的方法( $t = -1.0$  s)

## 4 讨论和总结

计算表明,本文的计算方案对 level\_set 函数的重新初始化在数值上是有效的。它比较准确地捕捉到界面位置,能够较好地保持流体体积。由于界面位置是由 level\_set 函数零等值线和网格线的交点来确定,位置的精度与插值方法有关,可以采用高阶的插值方法来获得界面位置的保持能力。表面上看,算法中增加了插值步骤,而且这种计算随着界面的复杂程度线性增加。但由于算法的特性,速度上的收益部分抵消了增加该步骤的损失。本工作提供了一种新的计算策略,可以为 level\_set 函数的重新初始化计算提供一种新的实现途径,在一定程度上解决了困扰 level\_set 方法体积不守恒的问题,不过应该指出该方法由于没有直接对体积进行保持处理,必然会存在体积的损失或波动。

## [参考文献]

- [1] Osher S, Sethian J A. Fronts propagating with curvature dependent speed: algorithms based on

- Hamilton-Jacobi formulations [J]. *J. Comput Phys*, 1988, **79**(1): 12—37.
- [2] Mulder W, Osher S, Sethian J A. Computing interface motion in compressible gas dynamics [J]. *J. Comput Phys*, 1992, **100**(1): 209—228.
- [3] Fatemi E, Sussman M. An efficient interface preserving level\_set re\_distancing algorithm and its application to interfacial incompressible fluid flow [J]. *SIAM J Sci Statist Comput*, 1995, **158**(1): 36—58.
- [4] Sussman M, Smereka P. Axisymmetric free boundary problems [J]. *J. Fluid Mech*, 1997, **341**(1): 269—294.
- [5] Sussman M, Almgren A S, Bell J B, et al. An adaptive level\_set approach for incompressible two-phase flows [J]. *J. Comput Phys*, 1999, **148**(1): 81—124.
- [6] Chang Y C, Hou Y T, Merriman B, et al. A level\_set formulation of Eulerian interface capturing methods for incompressible fluid flows [J]. *J. Comput Phys*, 1996, **124**(2): 449—464.
- [7] Chen S, Merriman B, Osher S, et al. A simple level\_set method for solving Stefan problem [J]. *J. Comput Phys*, 1997, **135**(1): 8—29.
- [8] Fedkiw R P, Aslam T, Merriman B, et al. A nonoscillatory Eulerian approach to interface in multimerial flows (the ghost fluid method) [J]. *J. Comput Phys*, 1999, **152**(2): 457—492.
- [9] Jiang G S, Shu C W. Efficient implementation of weighted ENO schemes [J]. *J. Comput Phys*, 1996, **126**(1): 202—228.
- [10] 刘儒勋, 王志峰. 数值模拟方法和运动界面追踪 [M]. 合肥: 中国科学技术大学出版社, 2001.

## An Improved Level\_Set Re\_Initialization Solver

WANG Zhi\_liang, ZHOU Zhe\_wei

(Shanghai Institute of Applied Mathematics and Mechanics,  
Shanghai University, Shanghai 200072, P. R. China)

**Abstract:** Re\_initialization procedure in level\_set interface capturing method were investigated. The algorithm accomplishes the re\_initialization step through locking the interface positions. Better accuracy was obtained both on the interface positions and the total fluid volume keeping. Though one more step of the interpolations is added in the procedure, there is no significant increase in total machine time spent.

**Key words:** level set method; volume conservation; level set re\_initialization