不可压流体的模拟、渲染和应用研究

摘 要

不可压流体

**关键词：**计算机图形学，拉式模拟，OpenGL，屏幕空间渲染，不可压流体，实时应用，CUDA

**The simulation, rendering and application of incompressible fluids**

**ABSTRACT**

**Key words：**computer graphics, Lagrangian simulation, OpenGL, screen-space rendering, incompressible fluids，real-time application， CUDA

目 录

[1 引 言 1](#_Toc402184259)

[1.1 研究背景 1](#_Toc402184260)

[1.2 流体模拟的研究现状与相关工作 1](#_Toc402184261)

[1.3 流体渲染的研究现状与相关工作 1](#_Toc402184264)

[1.4 本文所作的工作 1](#_Toc402184265)

[2 流体的PBF模拟方法 2](#_Toc402184266)

[2.1 Navier-Stokes方程组 2](#_Toc402184267)

[2.2 平滑粒子动力学模型 2](#_Toc402184268)

[2.3 求解不可压性的PBF方法 2](#_Toc402184268)

[2.3.1 流体位置的压强修正 2](#_Toc402184269)

[2.3.2 流体速度的涡度、粘度修正 2](#_Toc402184270)

[2.4 PBF方法的GPU并行化实现 2](#_Toc402184268)

[2.4.1 CUDA并行编程简介 2](#_Toc402184269)

[2.4.2 近邻粒子寻找算法 2](#_Toc402184269)

[2.4.3 实现管线细节 2](#_Toc402184269)

[3 流体的屏幕空间渲染 3](#_Toc402184272)

[4 结果和讨论 4](#_Toc402184280)

[5 结论和展望 6](#_Toc402184288)

[5.1 结论 6](#_Toc402184289)

[5.2 展望 6](#_Toc402184290)

[参考文献 7](#_Toc402184291)

[谢 辞 8](#_Toc402184292)

# 1 引 言

## 1.1 研究背景

计算流体力学（Computational Fluid Dynamics, CFD）是使用数值方法模拟流体运动的学科，流体模拟对象包括水流、火焰、烟雾、流动的沙粒等一系列自然现象和景观。航空航天、传播制造等领域对模拟的物理真实性有很高要求。与这些领域不同，计算机图形学中常常牺牲一定的物理准确性，以较小的计算量，追求视觉上的真实性。计算机游戏、医疗研究等交互性较强的应用，更是对流体的模拟和渲染提出了严格的实时性要求。由于传统的模拟方法计算量庞大，并行化程度低，不能很好满足实时性要求，流体模拟的实时化逐渐成为业界和学界研究的热点。

## 1.2 流体模拟的研究现状与相关工作

根据微分方程离散方式的不同，模拟方法通常可分为欧式法（Euler method）、拉式法（Lagrangian method）和混合方法（Hybrid method）三类。本世纪初Stam等人发表的两篇重要论文[1][2]，成为了欧式法的奠基之作。欧式法将空间分割为离散的格子（Grid），将流体的速度、压强、温度等物理参数记录在格子中。通过将NS方程进行有限差分，欧式法将微分方程求解转化为高阶线性方程的求解。针对欧式法均匀采样浪费空间，损失精度的问题，Losassao提出了八叉树[3]，Feldman[4]等人提出了非结构四面体网格（Unstructured tetrahedral meshes）的离散方法。然而，这些方法的实现十分复杂，同时精度通常逊于朴素的欧式方法。传统欧式法依旧存在过高内存要求、采样精度损失和数值耗散等一些问题[5]。2016年，Chern等人发表了论文《薛定谔的烟》[6]，提出使用为值域的波函数描述流体状态，并使用凝聚态物理中用于求解超流体的非线性薛定谔方程（Gross-Pitaevskii方程）作为流体的动力学方程，开发了一种没有数值耗散，同时可以高效求解其不可压性的欧式方法。这种方法在对烟雾的模拟上取得了突破性的效果。但对于需要渲染表面的水流等液体，目前并没有好的波函数表面重建方法。

拉式法将流体离散为固定数量，在约束下自由活动的粒子，由粒子记录流体状态的各个物理量。拉式法可追溯到20世纪70年代Gingold、Lucy等人在研究星际气体运动的工作[7][8]。Desbrun和Cani[9]最先把Gingold的平滑粒子流体动力学方法（Smoothed Particle Hydrodynamics, SPH）引入图形学中。随后，Müller的工作[10]展现了SPH在实时动画和交互应用中的前景。Solenthaler等人提出的预测校正不可压SPH（Predictive-corrective incompressible，SPH）第一次实现了满足不可压条件的拉式方法，将拉式方法推进了一大步，并在实时模拟领域得到了广泛的应用。随后Macklin等人的基于位置的流体（Position base fluids），又在同等精度下提高了模拟步长和稳定性，进一步提升了流体模拟的实时性。该方法是本文实现的重点。

混合方法是目前电影工业界常用的，精度最高的方法。混合方法使用欧式法求解不可压方程，欧式法精度高且易于处理流体边界；使用拉式法处理流体的对流（advection），即各物理量随着流体在速度场的转移，从而避免了欧式法的数值耗散。经典的混合方法包括八十年代流行至今的粒子元胞法（Particle-In-Cell, PIC）[11]，和隐式粒子流体法（FLuid Implicit Particle, FLIP）[12]。商业软件如Houdini，Realflow几乎都是使用FLIP方法。最近Jiang等人提出了仿射粒子元胞法（Affine Particle-In-Cell, APIC）[13]，其兼具了PIC的稳定性和FLIP低耗散的特点。其改进，Fu提出的多项式粒子元胞法（A polynomial particle-in-cell method）[14]，大幅减少了能量和旋量(vorticity)在数值计算中的损失。APIC被已经应用在迪士尼和皮克斯最新的电影中（如《海洋奇缘》）[15]。

## [1.3 流体渲染的研究现状与相关工作](#生成函数法及其优势)

## 1.4 本文所作的工作

本文针对不可压流体，实现了一种新的拉式（Lagrangian）实时模拟方法，称为基于位置的流体（Position based fluids）方法[16]。同时，本文实现了一种屏幕空间的拉式流体渲染方法，这种方法利用深度缓存（Depth buffer）还原粒子化表示的液体[17]，省去了传统方法中的重建步骤，表现出优秀的实时效果。

# 2 流体的PBF模拟方法

一切计算机物理模拟都是对描述物理现象的物理方程的有限近似和数值求解。本节首先从流体的物理性质出发，导出准确描述流体的Navier-Stokes方程，然后介绍了本文使用的空间离散方法——平滑粒子动力学模型。平滑粒子动力学模型是拉式模拟的一种典型实现方法，但它本身不描述流体的运动规律。本节接着引入了Navier-Stokes方程的时间离散方法——“基于位置的流体(Position Based Fluids, PBF)”方法。这种方法最突出贡献，是使用雅可比方法迭代修正不可压方程，从而以很小的代价、优越的强健性解决了流体不可压约束的问题。本节最后描述了使用CUDA的PBF方法并行实现。

## 2.1 Navier-Stokes方程组

Navier-Stokes方程组是描述流体运动规律的方程组。本文研究的流体满足不可压条件和牛顿条件两个条件。后文将详细描述两个条件和它们的使用范围。从这两个条件出发，本节导出Navier-Stokes方程组的完整形式。

（图1(a), (b)）

首先我们考虑空间中的一块流体区域，这块区域内的任意一点有密度、速度等物理量。设外力为，对这一块流体区域应用**牛顿第二定律**，有

注意（1）中研究的对象是一块固定的流体，含有固定的一些分子。由于流体会流动，区域是随时间变化的量。（1）式的左侧可以用**莱布尼茨法则**展开。莱布尼茨法能够将对积分的导数转化成对导数的积分，如下所示，是液体的任意一个物理量，是区域的边界，有

其中方程右侧 是应用了三维空间中的散度定理。

莱布尼茨法则有一个显著的物理意义。方程左侧物理量在整块区域的变化，是右侧第一项区域内随时间变化，与右侧第二项体积变化叠加的结果。

将莱布尼茨法则应用到（1）左侧，可以得到

上式比较复杂，但实际上，我们可以利用质量守恒定律做简化。空间虽然会随时间发生变动，但我们考虑的其中的流体既不会增加也不会减少，因此有**质量守恒定律**

对质量守恒定律应用莱布尼茨法则，有

因为对任意的上面的等式都成立，必有，而这一项恰好出现在（3）中，因此可将（3）简化。将简化的（3）带回（1），得到牛顿定律导出的最终方程，文献中常称之为**动量方程**。

如果将（5）与牛顿第二定律方程的基本形式相比较，能够发现两者之间形式上的相似性：对应，同时中的正好是加速度的定义，而多出来的一项是“漂移“产生加速度。为了便于理解，考虑一个具体的例子：假设有一条河流，河面上每个位置的水流速度是固定的，但是不同位置的速度是不同的。当一个人泛舟而下时，假设船速与水流速度相等，他将会感受到船的加速度，这个加速度是因为船经过了速度不同的区域。中表征了速度变化的方向，而则表征了漂移的方向。

为了简化表示，定义**材料微分**运算符

利用（6）将（5）重写为（7）

现在考虑作用在体积上的外力，它们的存在使得液体的速度发生改变。首先容易想到，应包含有重力，其中是重力常数。除此之外，流体会还受到周围流体的挤压，产生压强力。对于有粘性的流体，还会因为与周围流体有流速差异受到粘性力。下面我们将逐项分析。

首先考虑压强力。压强仅在体积的表面对有贡献，因为内部的压强力将会互相抵消。对于流体，通常可以认为压强是各向同性的，即一点处的压强对四周任意方向施加的压强力大小是相同的，因此压强是一个标量。

## 2.2 平滑粒子动力学模型

### [2.2.1 直接计算生成函数法](#直接计算生成函数法测定稳定常数的原理)

……。

### 2.2.2 分段拟合生成函数法

……。

### 2.2.3 半整数生成函数法

……。

## 2.3 求解不可压性的PBF方法

## 2.4 PBF方法的GPU并行化实现

# 3 实验部分

## 3.1 仪器和试剂

### 3.1.1 仪器

798-MPT全自动电位滴定仪（瑞士万通）；

氢离子选择性复合电极（瑞士万通）；

……。

### 3.1.2 试剂

氢氧化钠（A.R.）；

邻苯二甲酸氢钾（容量基准试剂）；

氯化钾（A.R.）；

……。

## 3.2 溶液的配制及标定

### 3.2.1 NaOH标准溶液的配制及标定

称12.0g 固体NaOH至100mL烧杯，蒸馏水溶解，转移至试剂瓶，用蒸馏水稀释至总体积为3L，待标定。

将基准邻苯二甲酸氢钾于105oC～110 oC烘至恒重（约3小时），置于干燥器，冷却至室温。精确称取基准邻苯二甲酸氢钾若干（见表3.1）至100mL烧杯，分别加入50mL蒸馏水，溶解，用待标定的NaOH溶液自动电位滴定，平行实验三次。NaOH标准溶液的标定结果见表3.1。

表3.1 氢氧化钠溶液浓度标定

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 邻苯二甲酸氢钾质量/g | NaOH溶液体积/mL | NaOH溶液浓度/（mol·L-1） |
| 1 | 0.1118 | 5.91 | 0.0926 |
| 2 | 0.1028 | 5.42 | 0.0929 |
| 3 | 0.1124 | 5.91 | 0.0932 |

### 

### 3.2.2 氯化钾离子强度调节剂的配制

……。

## 3.3 实验步骤

……。

# 4 结果和讨论

## 4.1 多元酸体系的结果和讨论

### 4.1.1 直接计算法

……。

### 4.1.2 半整数法

A. 半整数法的求解过程

a. 可以直接得到半整数的情况



从表4.1可见，当为0.5和1.5时的pH分别是5.32和2.74，即……。



b. 无法直接得到半整数的情况



在用半整数生成函数法直接求解时，会遇到这样的问题……。表4.4列出了滴定过程中乙二酸溶液的各主要物理量的部分数据。从表4.4可见，……。

……

……。

表4.4 草酸的部分数据列表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *V*/mL | *E*/mV | pH | [H+]/（mol·L-1） |  |
| …… | …… | …… | …… | …… |
|  |  |  |  |  |
| 1.00 | 2.62×102 | 2.22 | 6.07×10-3 | 1.14 |
| 1.50 | 2.60×102 | 2.27 | 5.42×10-3 | 1.10 |
|  |  |  |  |  |
| …… | …… | …… | …… | …… |

续表4.4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *V*/mL | *E*/mV | pH | [H+]/（mol·L-1） |  |
| …… | …… | …… | …… | …… |
| …… | …… | …… | …… | …… |
| …… | …… | …… | …… | …… |



图4.1 乙二酸 ≥1.15数据段曲线及其拟合曲线（实线--实际曲线，虚线--拟合曲线）



针对这种情况，可以采用多项式拟合的方法求解。以乙二酸为例，在Excel中，选取1.28<<1.15之间的数据，以为横坐标，pH为纵坐标，做-pH曲线（见图4.1），并添加趋势线，选择相关系数*R*2最接近1的多项式作为拟合方程，……。



B. 半整数法的计算结果

利用半整数生成函数法，对各种多元酸三次平行实验数据分别进行处理，并求平均值，结果见……。

C. 半整数法计算结果的讨论

……。

### 4.1.3 分段拟合法

……。

## 4.2 氨基酸合铜体系的结果和讨论

……。

## 4.3 关于计算方法的讨论

## 4.4 关于其他问题的讨论

# 5 结论和展望

## 5.1 结论

（1）生成函数法可以分为直接计算生成函数法、分段拟合生成函数法及半整数生成函数法。这三种方法有如下特点：①……；②……；③……。

（2）本文运用三种不同生成函数法，测定了多元酸和氨基酸合铜配合物的稳定常数，得到了……。

（3）三种生成函数法中无论哪一种方法，对待测酸或配合物稳定常数的大小均有一定的要求，如……。

……。

## 5.2 展望

（1）生成函数法理论可靠，计算方便，但……。

（2）在生成函数法的应用中，还有以下问题有待研究和解决：①……；②……。

……。

# 

# 参考文献

[1] J. Stam, “Stable fluids,” in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques - SIGGRAPH ’99*, 1999, pp. 121–128.

[2] R. Fedkiw, J. Stam, and H. W. Jensen, “Visual simulation of smoke,” in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques - SIGGRAPH ’01*, 2001, pp. 15–22.

[3] F. Losasso, F. Gibou, and R. Fedkiw, “Simulating water and smoke with an octree data structure,” in *ACM SIGGRAPH 2004 Papers on - SIGGRAPH ’04*, 2004, vol. 23, no. 3, p. 457.

[4] B. E. Feldman, J. F. O’Brien, B. M. Klingner, and T. G. Goktekin, “Fluids in deforming meshes,” in *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation - SCA ’05*, 2005, p. 255.

[5] R. Bridson, *Fluid Simulation for Computer Graphics*. A K Peters, 2007.

[6] A. Chern, F. Knöppel, U. Pinkall, P. Schröder, and S. Weißmann, “Schrödinger’s smoke,” *ACM Trans. Graph.*, vol. 35, no. 4, pp. 1–13, Jul. 2016.

[7] R. A. Gingold and J. J. Monaghan, “Smoothed particle hydrodynamics: theory and application to non-spherical stars,” *Mon. Not. R. Astron. Soc.*, vol. 181, no. 3, pp. 375–389, Dec. 1977.

[8] L. B. Lucy, “A numerical approach to the testing of the fission hypothesis,” *Astron. J.*, vol. 82, p. 1013, Dec. 1977.

[9] M. Desbrun and M.-P. Gascuel, “Smoothed Particles: A new paradigm for animating highly deformable bodies,” Springer, Vienna, 1996, pp. 61–76.

[10] M. Müller, D. Charypar, and M. Gross, “Particle-Based Fluid Simulation for Interactive Applications,” *Proc. 2003 ACM SIGGRAPH/Eurographics Symp. Comput. Animat.*, no. 5, pp. 154–159, 2003.

[11] F. H. Harlow, “The particle-in-cell method for numerical solution of problems in fluid dynamics,” in *Proceedings of Symposia in Applied Mathematics*, 1963.

[12] J. U. Brackbill and H. M. Ruppel, “FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions,” *J. Comput. Phys.*, vol. 65, no. 2, pp. 314–343, Aug. 1986.

[13] C. Jiang, C. Schroeder, A. Selle, J. Teran, and A. Stomakhin, “The affine particle-in-cell method,” *ACM Trans. Graph.*, vol. 34, no. 4, p. 51:1-51:10, Jul. 2015.

[14] C. Fu, Q. Guo, T. Gast, C. Jiang, and J. Teran, “A polynomial particle-in-cell method,” *ACM Trans. Graph.*, vol. 36, no. 6, pp. 1–12, Nov. 2017.

[15] Stuart Wolpert, “UCLA mathematicians bring ocean to life for Disney’s ‘Moana’ | UCLA,” 2017. [Online]. Available: http://newsroom.ucla.edu/stories/ucla-mathematicians-help-bring-the-ocean-to-life-for-disneys-hit-movie-moana. [Accessed: 14-Mar-2018].

[16] M. Macklin and M. Müller, “Position based fluids,” *ACM Trans. Graph.*, vol. 32, no. 4, p. 1, Jul. 2013.

[17] W. J. van der Laan, S. Green, and M. Sainz, “Screen space fluid rendering with curvature flow,” in *Proceedings of the 2009 symposium on Interactive 3D graphics and games - I3D ’09*, 2009, p. 91.

# 谢 辞

正文内容