



北京科技大学
University of Science and Technology Beijing

密级：公开

本科生毕业设计(论文)

题 目：基于物理的 SPH 非牛顿流体仿真

作 者：申龙

学 号：41924152

学 院：计算机与通信工程学院

专 业：计算机科学与技术

成 绩：_____

2023 年 06 月

本科生毕业设计(论文)

题 目: 基于物理的 SPH 非牛顿流体仿真

英文题目: Physics based SPH simulation

Of non-Newtonian fluid

学 院: 计算机与通信工程学院

班 级: 计 193

学 生: 申龙

学 号: 41924152

指导教师: 张雅斓 职称: 讲师

指导教师: 职称:

声 明

本人郑重声明：所呈交的论文是本人在指导教师的指导下进行的研究工作及取得研究成果。论文在引用他人已经发表或撰写的研究成果时，已经作了明确的标识；除此之外，论文中不包括其他人已经发表或撰写的研究成果，均为独立完成。其他同志对本文所做的任何贡献均已在论文中做了明确的说明并表达了谢意。

学生签名：

申龙

2023 年 05 月 29 日

导师签名：

张雅坤

2023 年 05 月 29 日

毕 业 设 计 (论 文) 任 务 书

一、学生姓名：申龙 学号：41924152

二、题目：基于物理的 SPH 非牛顿流体仿真

三、题目来源：真实 ☒ 、 自拟 ☐

四、结业方式：设计 ☒ 、 论文 ☐

五、主要内容：

1. 查阅国内外相关文献，了解非牛顿流体的物理仿真方法。

2. 基于 SPH 方法实现非牛顿流体模拟。

六、主要（技术）要求：

1. 使用 Taichi 语言，设计物理模拟框架。

2. 采用 DFSPH 进行流体模拟。

3. 采用 Cross 模型统一建模非牛顿流体。

4. 采用合适的热传导模型建模温度相关的粘度模型。

七、日程安排：

第 1 周：确定毕业论文内容，了解相关知识。

第 2-4 周：查阅国内外相关文献，熟悉 SPH 物理模拟方法，完成选题报告。

第 5-10 周：学习现在先进的 SPH 算法，编程实现 SPH 模拟框架，完善粘度求解器，嵌入 Cross 模型建模非牛顿流体。撰写并提交中期检查表，准备中期答辩。

第 11-12 周：嵌入温度-粘度模型，实现融化等流体现象。

[1] 第 13-15 周：对毕业设计进行总结，撰写毕业论文，完成毕业设计答辩。

八、主要参考文献和书目：

[1] Bender J, Koschier D. Divergence-free smoothed particle hydrodynamics[C]. //Proceedings of the 14th ACM SIGGRAPH/Eurographics symposium on computer animation. 2015: 147-155.

[2] 刘树森, 何小伟, 王文成, 吴恩华. 光滑粒子流体动力学 (SPH) 流体仿真技术综述. 软件学报. 2022.

[3] Shusen Liu, Xiaowei He, Wencheng Wang, Enhua Wu. "Adapted SIMPLE Algorithm for Incompressible SPH Fluids With a Broad Range Viscosity". IEEE Transactions on Visualization and Computer Graphics. 2022, 28(9): 3168-3179.

[4] 张雅斓. 基于物理及数据驱动的非牛顿流体动画[D]. 北京科技大学, 2020. DOI:10.26945/d.cnki.gbjku.2020.000252.

- [5] Luiz Fernando de Souza andrade, Marcos Sandim, Afonso Paiva. “SPH Fluids for Viscous Jet Buckling”. IEEE SIBGRAPI Conference on Graphics, Patterns and Images. 2014, 65 - 72.
- [6] B.Solenthaler and R.Pajarola, “Predictive-corrective incompressible SPH”. ACM Trans. on Graphics (SIGGRAPH Proc.). 2009, 40:1 - 40:6.
- [7] Dan Koschier, Jan Bender, Barbara Solenthaler, and Matthias Teschner. “Smoothed particle hydrodynamics techniques for the physics based simulation of fluids and solids”. Eurographics Tutorial. 2019.
- [8] M. Weiler, D. Koschier, M. Brand, and J. Bender. “A physically consistent implicit viscosity solver for SPH fluids”. Comput. Graph. Forum. 2018, 37: 145-155.

指导教师签字:

张雅坤

2023 年 2 月 23 日

学 生 签 字:

申龙

2023 年 2 月 23 日

系（所）负责人章:

年 月 日

摘 要

计算机图形学是链接计算机与真实世界的重要桥梁。在现实的生产生活中，人们已离不开计算机。计算机图形学的任务是帮助人们在电子世界中构建真实世界，模拟物理现象。对真实场景模拟的需求也一直广泛存在于各行各业之中。

模拟流体运动一直是该领域的研究热点。流体广泛存在于大自然中，种类繁多且特性不同。当前，基于物理的流体模拟技术的研究主要聚焦于牛顿流体，例如标准条件下的水、空气等，其粘度在一定条件下是定常的。但与此截然不同的是非牛顿流体，其粘度是关于剪切率的非线性函数，这种流体在工业生产和自然界中更加常见，如血液、巧克力、油漆、玻璃熔融物等。

本文研究并实现了一种基于 SPH 的非牛顿流体的物理模拟方法。该方法首先采用 Cross 模型对非牛顿流体的变粘特性建模，然后采用基于热传导方程的温度-粘度模型为粘度增加温度属性，最后采用 DFSPH 进行仿真优化。标准 DFSPH 方法往往着眼于压力计算，本文结合 DFSPH 的优势，在算法中添加了 Cross 变粘模型和温度变粘模型来分别实现非牛顿流体模拟和受温度影响的流动现象。

本文的方法能够实现剪切变稀型、剪切增稠型流体的模拟以及高温加速融化等流体现象。

关键词： 计算机图形学，物理模拟，非牛顿流体，光滑粒子流体动力学

Physics based SPH simulation of non-Newtonian fluid

Abstract

Computer Graphics is a crucial bridge that links the computer with the real world. In real-life production and living, people are inseparable from computers. The mission of computer graphics is to help people construct the real world in the electronic realm, simulating physical phenomena. The demand for simulating real scenarios persists widely in various industries.

Simulating fluid motion has always been a research hotspot in the field. Fluids exist widely in nature, and there are many kinds of fluids with different properties. At present, the research on fluid animation based on physics mainly focuses on Newtonian fluid, which is a fluid that obeys Newton's law of viscosity, such as water and air under standard conditions, and its viscosity is constant under certain conditions. But in contrast to this are non-Newtonian fluids, whose viscosity is a nonlinear function of shear rate, which are more common in industrial production and nature, such as blood, chocolate, paint, glass melt, etc.

This thesis studies and realizes a physical simulation method of non-Newtonian fluid based on SPH. The method first uses the Cross model to model the viscous properties of non-Newtonian fluids, then uses the temperature-viscosity model based on the heat conduction equation to add temperature attributes to the viscosity, and finally uses DFSPH for simulation optimization. Standard DFSPH methods often focus on pressure calculations. In this thesis, combining the advantages of DFSPH, a Cross viscous model and a temperature viscous model are added to the algorithm to realize non-Newtonian fluid simulation and flow phenomena affected by temperature, respectively.

The method in this thesis can realize the simulation of shear-thinning and shear-thickening fluids, as well as high-temperature accelerated melting and other fluid phenomena.

Key Words : Computer Graphics , Physical Simulation , non-Newtonian fluid , SPH

目 录

摘 要	I
Abstract	III
1 引 言	1
1.1 课题背景及意义	1
1.2 非牛顿流体仿真研究现状	2
1.3 本文研究内容及创新点	4
1.4 本文组织结构	4
2 物理模拟与 SPH 方法概述	6
2.1 基于物理的模拟	6
2.1.1 欧拉模拟	6
2.1.2 拉格朗日模拟	8
2.2 SPH 方法概述	9
2.2.1 SPH 离散化	9
2.2.2 邻居搜索	11
2.2.3 不可压缩求解器	12
2.2.4 粘度求解器	13
2.2.5 表面张力	14
2.2.6 边界处理	15
2.3 概念小结	17
3 基于 Cross 模型的非牛顿流体仿真	19
3.1 剪切变稀与剪切增稠型流体	19
3.2 广义牛顿流体	20
3.3 算法实现	22
4 基于温度-粘度模型的非牛顿流体仿真	24
4.1 基于热传导方程的温度-粘度模型	24
4.2 流体自由表面检测	25
4.3 算法实现	26
5 仿真验证与优化	27
5.1 基于 SESPH 的算法验证	27
5.2 使用 DFSPH 优化仿真	28
6 实验与分析	32
6.1 CFL 条件设置	32

6.2 Cross 模型参数对流体类型的影响	33
6.3 温度-粘度模型实验	37
6.4 DFSPH 性能对比	40
7 总结与展望	42
参考文献	43
在学取得成果	47
致 谢	49

1 引 言

1.1 课题背景及意义

计算机图形学是链接计算机与真实世界的重要桥梁。在现实的生产生活中，人们已离不开计算机。计算机图形学的任务是帮助人们在电子世界中构建真实世界，模拟物理现象。对真实场景模拟的需求也一直广泛存在于各行各业之中，其中包括教育教学、工业生产、军事模拟和电商等领域。在电影业中，为了实现更加宏大且不可人为复现的场景，如海啸冲击、火山喷发、山体滑坡等，均需要计算机动画技术对其进行模拟。另外，计算机动画还能将艺术家们天马行空的想象变成现实，如星际穿越、未来战争等。

随着技术的发展，人们对所搭建场景的真实度和沉浸感要求日益增加，若模拟由纯美工来实现，则会产生不真实感，对大场景的构建也将会是一个难题，因此基于物理的模拟至关重要。受制于计算机的性能，基于物理的计算机模拟往往会在精细度和高效模拟之间做权衡。

模拟流体运动一直是该领域的研究热点。流体广泛存在于大自然中，其种类繁多且特性不同。当前，基于物理的流体模拟技术的研究主要聚焦于牛顿流体，例如标准条件下的水、空气等，其粘度在一定条件下是定常的。但与此截然不同的是非牛顿流体，其粘度是关于剪切率的非线性函数，这种流体在工业生产和自然界中更加常见，如血液、巧克力、油漆、玻璃熔融物等。也就是说非牛顿流体更加普遍，而牛顿流体则是苛刻条件下的产物。因此，对非牛顿流体的模拟既能够补充传统流体动画的不足，也是图形学等相关领域的一个非常重要的课题。

对于物理仿真，目前已有多种数值模拟方法，例如欧拉法^{[1] [2] [3]}，拉格朗日法^{[4] [5] [6] [7]}，混合方法^{[8] [9] [10]}等。其中，SPH 方法（Smoothed Particle Hydrodynamics，光滑粒子流体动力学）近来在物理模拟中愈发流行，它是一种拉格朗日方法，其对比其他方法有如下几点优势：

- 相比于欧拉法，SPH 方法天然满足材料边界的细致划分。由于流体的运动现象通过 SPH 粒子的位置变化体现，SPH 粒子的计算又相互独立，因此处于自由表面的粒子的运动特征更易于被捕获，这也使得 SPH 方法能够很好的处理多材料的耦合问题。
- 相比于欧拉法和混分方法，SPH 方法的计算框架和编程实现较为简

单，且 SPH 框架非常灵活，能够同其他方法结合优化数值误差、视觉伪影等问题。

- SPH 作为无网格的拉格朗日方法，其天然满足质量守恒定律。
- 此外，SPH 方法适用的模拟场景非常广泛，包括流体、弹塑性固体、沼泽地、雪地、海绵等，由于 SPH 粒子的属性独立计算，所以也能很好的模拟耦合、热传导等现象。

尽管 SPH 方法有诸多优点，但 SPH 离散化也会产生一些问题。例如计算效率不高，多种类型的数值计算不稳定等。

针对以上需求和问题，本课题致力于构建一个基于 SPH 方法的高效、准确、物理的非牛顿流体仿真系统，实现逼真的非牛顿流体动画。

1.2 非牛顿流体仿真研究现状

在低速场景下，流体可以看作是分层流动的，即层流模型（如图 1-1）。当两层相邻的流层发生相对运动时，会产生滑动摩擦力，这种摩擦力体现出的阻碍流层运动的性质称为粘性，这种摩擦力则称为粘滞力。对于牛顿流体，其粘度在一定条件下为常数；而非牛顿流体的粘度是动态变化的。将粘度定义为剪切力与速度梯度的比例系数，则可知非牛顿流体的粘度模型是非线性的。现实世界中存在着大量的非牛顿流体，但长期以来，基于物理的模拟对象多聚焦于牛顿流体。对于传统流体模拟来说，如何快速而真实地仿真非牛顿流体动画是一个具有挑战性的问题。



图 1-1 层流模型

非牛顿流体不服从牛顿粘性定律，即其应力与应变的关系是呈非线性的。对于非牛顿流体的仿真研究主要集中于如何模拟高粘流体，以及如何设计高效准确的 SPH 计算框架。

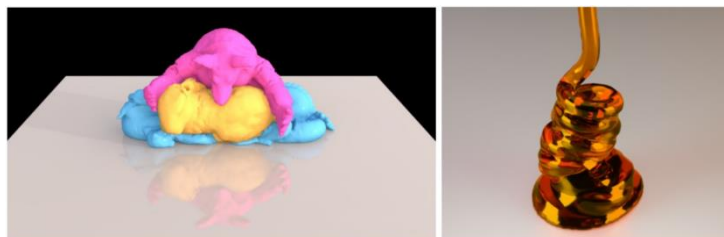


图 1-2 流体的粘性行为, 由左至右分别为文献^[16] ^[14] 中的配图

要得到准确、丰富的流体效果, 对非牛顿流体粘度的表达与计算至关重要。对 N-S 方程中粘度项的处理方式可分为人工粘度、基于粘度的拉普拉斯算子的方法和基于应变率张量的方法。

Schechter 等人^[11] 使用 XSPH 来计算粘度, 其通过人工参数来控制速度差, 以此来更新速度场, 但非物理的人工参数会导致无法正确模拟卷曲、盘绕等现象的问题。

对于粘度的拉普拉斯算子的处理有多种方式, 最直接的方法是对算子进行标准 SPH 离散化, 该方法计算便捷, 但是却对边界处的粒子缺失非常敏感, 同时也无法保证类高斯核函数的二阶导的符号正确性。为了避免求二阶导, Takahashi 等人^[12] 的方案中的相关项计算选择求解两次一阶导, 但导致计算开销增大。Brookshaw^[13] 提出使用一次 SPH 离散化结合一次有限差分来处理拉普拉斯算子, 以此保持线动量和角动量的守恒, Weiler 等人^[14]、Shusen Liu 等人^[15] 的方案中均采用了该方法。

为了更快、更稳定的模拟高粘流体, 许多研究提出了基于应变率张量的隐式粘度求解器。Takahashi 等人^[12] 基于最原始的应变率张量表达, 在反向欧拉框架下进行隐式时间积分, 该方案允许更大的时间步长, 但由于边界处的粒子缺失导致某些场景的仿真伪影问题。Bender 和 Koschier^[16] 提出了一个含应变率张量的速度约束函数矫正速度场, 但存在较大的计算开销。Peer 等人^[17] ^[18] 将速度梯度张量分解为三部分, 对其中的剪切速率张量施加了一个人工调控因子, 并使用 SPH 离散化和泰勒展开进行速度场的数值近似计算, 但为了平滑地重建速度场, 该方案不适合用于模拟低粘流体, 该速度场重建方法也会因为粒子缺失问题产生伪影。Weiler 等人^[14] 提出的基于拉普拉斯算子的优化方案解决了上述方法中的边界伪影问题。

为了进一步提高大粘性流体仿真的视觉真实性, 有研究者针对不可压缩性与粘性求解之间的干扰问题展开了研究。Shusen Liu 等人^[15] 借鉴欧拉网格法中的 SIMPLE 算法, 首次在 SPH 法中实现了可收敛的粘性-不可压缩性迭代算法, 减小了求解器间的干扰, 但算法的迭代部分可能引起线性增长的计

算开销。

1.3 本文研究内容及创新点

目前, 牛顿流体的模拟技术已经相对成熟, 在使用 SPH 方法进行非牛顿流体模拟时, 虽然可以沿用一些牛顿流体模拟的技术, 但还需要解决动态粘度、计算效率等问题, 这就需要拓展当前的 SPH 框架, 并选取合适的加速策略。

非牛顿流体的粘度模型不一, 需要选取更简洁且更稳定的方案。此外, 为了实现更加丰富的流体现象, 本文还将加入物理的温度-粘度模型, 实现流体块局部融化或温度相关的粘度变化现象。

以上, 本文的创新工作在于结合现有的 SPH 技术, 向现有算法中嵌入 Cross 模型和温度模型, 实现牛顿流体模拟框架对非牛顿流体模拟的拓展。本文的预期目标是实现对非牛顿流体中剪切变稀型流体、剪切增稠型流体以及温度相关的粘度变化现象的仿真。

1.4 本文组织结构

第一章主要讲述本文的课题背景, 并对研究现状进行概述, 指明本课题的实际意义以及具体技术创新。

第二章主要介绍相关的知识和技术, 具体从基于物理的模拟和 SPH 方法两方面展开。在基于物理的模拟章节中, 主要讲解现有的物理模拟概念、技术实现和优劣对比, 其次, 在光滑粒子流体动力学章节中, 将讲解本文所选择的 SPH 方法的核心概念和算法。最后给出本章节的小结。

第三章主要介绍本文提出并实现的基于 Cross 模型的非牛顿流体仿真算法。首先会介绍剪切变稀型和剪切增稠流体的特性, 其次介绍广义牛顿流体, 并分析对比现有模型的稳定性和表达能力, 最后给出本文算法实现。

第四章主要介绍本文提出并实现的基于温度-粘度模型的非牛顿仿真算法。首先介绍本文提出的基于热传导方程的温度-粘度模型, 其次介绍流体表面检测方法, 最后给出在 Cross 基础上进一步实现的温度-粘度模型算法。

第五章主要介绍仿真框架的完善与优化。首先介绍模型验证阶段采用的 WCSPH 方法, 其次介绍模型优化阶段采用的 DFSPH 方法。

第六章为本文的实验部分, 其中包括对 Cross 模型和温度-粘度模型的验证实验以及 SPH 方法的性能对比。

第七章为总结与展望。

2 物理模拟与 SPH 方法概述

2.1 基于物理的模拟

计算机图形学包含很多研究内容，近来随着动力学仿真和计算物理等技术理论日渐成熟，以及 3D 建模、渲染等技术的进步，基于物理的流体仿真得到不断发展，并逐渐成为人们研究的焦点之一。计算流体力学（Computational Fluid Dynamics，简称 CFD）是基于物理的流体仿真的基础。Navier-Stokes 方程（简称 N-S 方程）是流体力学中用来描述流体运动的控制方程，它是一组偏微分方程，但是可以简化它以适应计算机的计算。具体来说，先将空间进行离散化，再将 N-S 方程改写为离散化形式，最后使用数值模拟方法求解。一般来说，系统当前时刻的输入用来计算下一时刻的输出，如压强、密度等信息，通过这些信息更新流体的速度场和位置，并通过渲染模型逐帧渲染，以此得到流体流动的动画。

典型的不可压缩形式的 N-S 方程如下：

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho}\nabla p + \mu\nabla^2\mathbf{v} + \mathbf{f}_{ext} \quad (2-1)$$

$$\frac{d\rho}{dt} = -\rho\nabla\cdot\mathbf{v} = 0 \quad (2-2)$$

式 2-1 为 N-S 方程主体，其中， \mathbf{v} 为流体速度场； p 为流体压强场； μ 为流体运动粘度，是流体动力粘度与密度之比； \mathbf{f}_{ext} 为体积力加速度。该方程的含义是流体运动的加速度主要由不可压缩性引起的内部压强、有粘流体产生的内部剪切力和外部所受的体积力共同贡献而成。

式 2-2 为连续性方程，是流体模拟中的不可压缩条件与无散度条件，该方程表明，只要保证流体在任一时刻任一位置的速度场无散度，就能保证流体密度的恒定，从而满足不可压缩性。

理论上联立求解式 2-1 和式 2-2 即可实现基本的流体运动的数值模拟。但 N-S 方程是一个连续形式的微分方程，对其求解非常困难，因此需要进行适当的简化，使连续的微分方程转化为代数方程求解。

2.1.1 欧拉模拟

欧拉法将空间划分为相邻的网格，然后追踪流过每个网格的材料。在流体模拟中，空间被分隔为固定的网格，每个网格记录流过它的流体的速度、

密度等属性信息，这些属性值的变化由本身的变化及其周边介质的运动共同导致。因此，这里可以给出物质导数（material derivative）的定义：

$$\frac{Dq}{Dt} = \frac{\partial q}{\partial t} + \mathbf{v} \cdot \nabla q \quad (2-3)$$

其中 q 为任意的物理量。

式 2-1 是速度关于时间的全导数，而不是偏导数。在欧拉法中，一般使用另一种形式的 N-S 方程：

$$\frac{\partial \mathbf{v}}{\partial t} = -(\mathbf{v} \cdot \nabla) \mathbf{v} - \frac{1}{\rho} \nabla p + \mu \nabla^2 \mathbf{v} + \mathbf{f}_{ext} \quad (2-4)$$

并在数值模拟中，划分为对流项（advection）、外力项（body force）和投影项（pressure projection）进行求解：

$$\frac{Dq}{Dt} = 0 \quad (\text{advection}) \quad (2-5)$$

$$\frac{\partial \mathbf{v}}{\partial t} = \mathbf{f}_{ext} \quad (\text{body force}) \quad (2-6)$$

$$\frac{\partial \mathbf{v}}{\partial t} + \frac{1}{\rho} \nabla p = 0, \quad \nabla \cdot \mathbf{v} = 0 \quad (\text{pressure projection}) \quad (2-7)$$

通常，这三步的计算顺序没有要求，但对流项的求解要基于无散速度场的条件，因此，一般的欧拉模拟流程如表 2-1：

表 2-1 欧拉模拟的一般计算流程

算法：欧拉仿真流程	
01:	while run_sim do :
02:	初始化无散速度场 \mathbf{v}_{df}
03:	for 每个时间步 do :
04:	计算时间步长 Δt
05:	更新网格
06:	for 每个网格节点 c do :
07:	根据式 2-5 求解对流项获得 \mathbf{v}^1
08:	根据式 2-6 求解外力项获得 \mathbf{v}^2
09:	根据式 2-7 求解对流项获得 \mathbf{v}^{t+1}
10:	外推缓冲区域的速度场
11:	设定固定网格的速度
12:	for 每个粒子 i do :
13:	时间积分更新粒子位置

其中, \mathbf{v}^1 和 \mathbf{v}^2 是中间速度, \mathbf{v}^{t+1} 是最终的速度场。CFL 条件 (Courant-Friedrich-Lewy Condition) 用于约束时间步长的计算。

欧拉法是流体模拟的重要方法之一, 几十年中已取得长足的进步, 目前, 常见的欧拉法包括有限差分法^[19] (Finite Difference Methods, 简称 FDM), 有限体积法^[20] (Finite Volume Methods, 简称 FVM) 以及有限元法^[21] (Finite Element Methods, 简称 FEM) 等。

欧拉法多用于处理时变的连续介质模拟, 但由于其是把空间进行离散化, 所有信息由空间网格存储, 因此很难表达模拟的细节, 例如水花飞溅等场景。

2.1.2 拉格朗日模拟

与欧拉法不同的是, 拉格朗日法是将模拟对象离散化构成一个粒子系统, 每个粒子携带质量、速度、压强等属性信息。主流的拉格朗日法包括 SPH 法^{[22][23]}、基于位置的流体^[4] (Position Based Fluid, 简称 PBF) 等。SPH 方法近来被广泛使用于流体模拟之中, 并取得了众多成果。本文也将基于 SPH 方法进行研究实验。

拉格朗日法通常使用 N-S 方程的式 2-1 的形式。计算分为压力项、粘性项和外力项, 由于需要保证无散度条件, 因此三项的计算通常有顺序。SPH 方法中, 一般的计算流程如表 2-2 所示:

表 2-2 SPH 法的一般计算流程

算法: 经典 SPH 仿真流程	
01:	while run_sim do:
02:	其他操作
03:	for 每个粒子 i do:
04:	计算时间步长 Δt
05:	更新网格
06:	for 每个网格节点 c do:
07:	求解外力项获得 \mathbf{v}^1
08:	求解粘性项获得 \mathbf{v}^2
09:	求解压力项获得 \mathbf{v}^{t+1}
10:	时间积分更新粒子位置

相比于欧拉法，拉格朗日法多用于处理离散元素的模拟，该方法能够很好的捕捉自由表面的运动特征。但是拉格朗日法把模拟对象分隔成离散的元素，这是体积性质，所以通常很难保持，也就是说在流体模拟中很难保持一个恒定的密度或体积。

2.2 SPH 方法概述

SPH 方法将连续的流体介质离散化为独立的 SPH 粒子，这意味着流体的连续物理场可以通过数值方法进行计算，从而简化了微分方程的求解，该离散化是简单且有效的。

2.2.1 SPH 离散化

在物理学中，很多物理量定义为某种物理属性在空间上的积分，例如物体质量定义为体积质量密度在空间上的积分，即 $m = \int \rho(x) dv$ 。但对于质点来说，没有空间的概念，因此无法用这种方法定义其密度表达。狄拉克分布（Dirac - δ ）为此提供了思路，这是一种广义函数，并且满足归一化条件 $\int \delta(r) dv = 1$ ，其定义如下：

$$\delta(r) = \begin{cases} \infty, & r = 0 \\ 0, & \text{其他} \end{cases} \quad (2-8)$$

在质点场景中，它可以在质点处进行缩放。进一步的，可以将狄拉克分布用于基本离散化，则应用了狄拉克分布的连续紧支撑函数 $A(x)$ 等于其自身：

$$A(x) = (A * \delta)(x) = \int A(x') \delta(x - x') dv' \quad (2-9)$$

其中， dv' 表示对应于 x' 的积分变量。

然而，狄拉克分布不是函数且不能被离散化，因此需要一种方式对其进行近似。狄拉克分布实际上等于方差为零时的正态分布，因此可以选择高斯正态分布做积分近似，但因为其是非紧支撑域，所以最终选择一种更加通用的平滑函数 $W: \mathbb{R}^d \times \mathbb{R}^+ \rightarrow \mathbb{R}$ ，称为核函数（kernel functions）或平滑核（smoothing kernels）。因此，对于式 2-9 应用核函数 $W(\mathbf{r}, h)$ 如下：

$$A(x) \approx (A * W)(x) = \int A(x') W(x - x', h) dv' \quad (2-10)$$

其中， h 表示平滑长度，核函数的本质是一个加权，其应满足如下性质：

- (1) $\int_{\mathbb{R}^d} W(\mathbf{r}', h) dv' = 1$ （归一化条件）
- (2) $\lim_{h \rightarrow 0} W(\mathbf{r}', h) = \delta(\mathbf{r})$ （Dirac - δ 条件）

- (3) $W(\mathbf{r}, h) \geq 0$ (非负条件)
- (4) $W(\mathbf{r}, h) = W(-\mathbf{r}, h)$ (对称条件)
- (5) $W(\mathbf{r}, h) = 0$ for $|\mathbf{r}| \geq h$ (紧支撑条件)

对于平滑核的一个典型选择是三次样条核:

$$W(r, h) = \sigma_d \begin{cases} 6(q^3 - q^2) + 1, & \text{for } 0 \leq q \leq 0.5 \\ 2(1 - q)^3, & \text{for } 0.5 \leq q \leq 1 \\ 0, & \text{其他} \end{cases} \quad (2-11)$$

其中, $q = \frac{1}{h}|\mathbf{r}|$, 核归一化因子在不同维度下分别为 $\sigma_1 = \frac{4}{3h}$, $\sigma_2 = \frac{40}{7\pi h^2}$, $\sigma_3 = \frac{8}{\pi h^3}$ 。实际上, 存在着多种多样的核函数, 但该函数能够满足上述的所有要求 (如图 2-1), 并且拥有平滑长度与核函数支持半径相等的优势 ($h = \hbar$), 这有助于避免实现时的冲突。

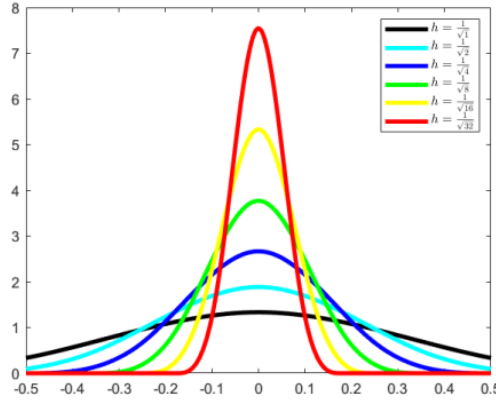


图 2-1 三次样条核函数

SPH 通常可以被理解作为一种用于空间场和空间微分运算的离散化方法。其本质是将空间中的物理场信息定义在 SPH 粒子上, 如流体密度、质量、速度、压强等, 它通过计算邻居粒子的贡献和核函数加权以确定当前粒子的下一时刻的状态信息。具体来说, 通过求和函数对式 2-9 进行连续近似可以得到以下重构估计来计算当前粒子处的场信息:

$$A_i = \sum_{j \in N(i)} \frac{m_j}{\rho_j} A_j W_{ij} \quad (2-12)$$

式 2-11 为标准 SPH 离散化, 其中, j 为粒子 i 的邻居粒子, $N(i)$ 表示 i 的邻域, 如图 2-2 所示。

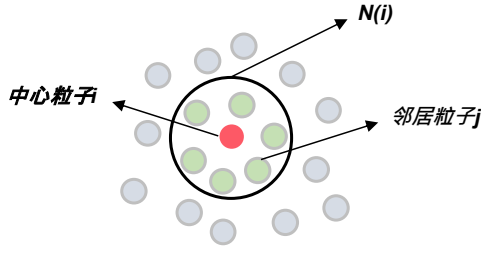


图 2-2 粒子的邻域

在流体的数值模拟中，常常会用到微分计算，借助标准 SPH 离散化，在求解微分时，只需要关注核函数，因此，场的散度和拉普拉子算子的计算如下：

$$\nabla \mathbf{A}_i = \sum_{j \in N(i)} \frac{m_j}{\rho_j} \mathbf{A}_j \otimes \nabla W_{ij} \quad (2-13)$$

$$\nabla \cdot \mathbf{A}_i = \sum_{j \in N(i)} \frac{m_j}{\rho_j} \mathbf{A}_j \nabla W_{ij} \quad (2-14)$$

$$\nabla^2 \mathbf{A}_i = \sum_{j \in N(i)} \frac{m_j}{\rho_j} \mathbf{A}_j \nabla^2 W_{ij} \quad (2-15)$$

式 2-13 中 $\mathbf{a} \otimes \mathbf{b} = \mathbf{ab}^T$ 。需要注意的是，如果直接将上面的式子用于数值计算，会出现违反牛顿第二定律的问题，即粒子 i 与粒子 j 的相互作用力不等。因此需要设计特殊形式的公式来解决这一问题，常见的有对称形式的公式^[24]。

2.2.2 邻居搜索

使用 SPH 离散化可以估计 SPH 粒子的各种物理属性，但在进行计算之前，需要先确定每个粒子的邻居。

一个粒子的邻居是该粒子邻域内的所有粒子的集合。通常，每个粒子的邻居需要在每一次迭代前重建。

邻居搜索的简单的方法是暴力搜索，其复杂度是 $O(n^2)$ ，通常来说，这是不可接受的。紧凑哈希表能够提供低复杂度的搜索能力，其基于背景网格，在三维场景中，背景网格将模拟空间分割为一个个相邻的元胞，如图 2-3 所示。

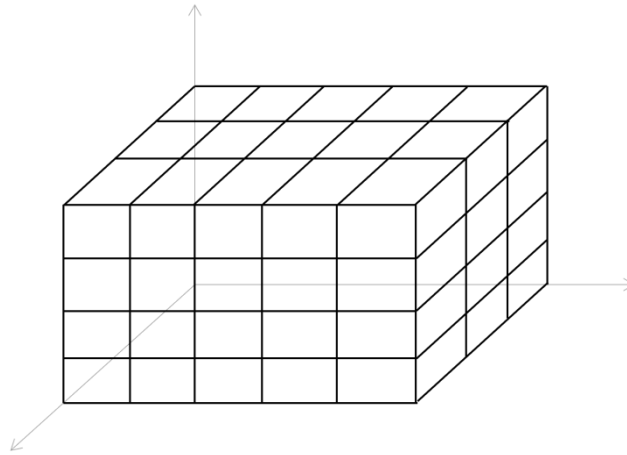


图 2-3 三维空间背景网格

在依赖背景网格做邻居搜索时，只需要搜索当前粒子所在的网格及邻居网格内的粒子即可。

2.2.3 不可压缩求解器

N-S 方程中的压力项主要用来保证流体体积不变以满足不可压缩性，通常将三项的计算划分为两类，先计算非压力项，如粘滞力、体积力等，来生成的中间速度，再对该速度场应用压力以矫正体积偏差。目前，已有多种不可压缩求解器方案。

State Equation SPH (SESPH) 是基于状态方程的 SPH 方法，密度偏差可以被显式地计算，也可以用微分形式计算。偏差项可以表示为实际密度与标准密度的比值或差值。例如 $p_i = k(\frac{\rho_i}{\rho_0} - 1)$, $p_i = k(\rho_i - \rho_0)$ 或 $p_i = k_1((\frac{\rho_i}{\rho_0})^{k_2} - 1)$ ，其中 ρ_0 为静止密度， k 为刚度参数，其是外力与型变量的比值，代表抵抗变形的能力。基于状态方程的 SPH 方法一般称为可压缩或者弱可压缩方法。Becker 和 Teschner^[25] 通过改进状态方程实现了弱可压缩流体模拟方案。相反，使用压力泊松方程 (Pressure Poisson Equation, 简称 PPE) 计算压力的方案是不可压缩方法，它通过求解一个线性系统来计算各自的压力场。

压力泊松方程是在流体动力学，特别是在不可压缩 (incompressible) 流体动力学中常用的一个方程。在这个背景下，PPE 用于找出能使流体速度场满足无散条件的压力场。它来自于 Navier-Stokes 方程的连续性方程和动量方程。

将通过非压力加速度生成的中间速度场表示为： $\mathbf{v}^* = \mathbf{v}(t) + \Delta t \mathbf{a}^{nonp}(t)$ ，接着利用连续性方程建立预估密度： $\rho^* = \rho(t) - \Delta t \nabla \cdot \mathbf{v}^*$ ，已知压力加速度的计算为 $-\frac{1}{\rho(t)} \nabla p(t)$ ，由此速度变化为 $-\Delta t \frac{1}{\rho(t)} \nabla p(t)$ ，该速度变化导致每时刻的密度变化，即 $-\rho(t) \nabla \cdot \left(\Delta t \frac{1}{\rho(t)} \nabla p(t) \right)$ ，该密度变化应抵消密度偏差 $\frac{\rho^0 - \rho^*}{\Delta t}$ 以保证密度恒定，即 $\frac{\rho^0 - \rho^*}{\Delta t} - \rho(t) \nabla \cdot \left(\Delta t \frac{1}{\rho(t)} \nabla p(t) \right) = 0$ 。这便是压力泊松方程的一个形式，通常写为式 2-16。

$$\Delta t \nabla^2 p(t) = \frac{\rho^0 - \rho^*}{\Delta t} \quad (2-16)$$

基于 PPE 的 SPH 方案被广泛研究，其能支持更大的时间步长。Solenthaler 和 Pajarola^[26] 通过预测-校正方案（Predictive-corrective incompressible SPH，简称 PCISPH）修正密度偏差，来实现不可压缩性。Ihmsen 等人^[27] 使用松弛雅可比方案隐式迭代求解 PPE（Implicit incompressible SPH，简称 IISPH）。但是，以上方法着重于修正密度偏差，并不能保证最终的速度场无散。因此，Bender 等人^[28] 提出了无散 SPH 方法（Divergence-Free SPH，简称 DFSPH），其不仅对密度进行修正，还约束速度场保持无散。

2.2.4 粘度求解器

基于 1.2 节的描述，粘度求解器可用人工粘度、基于应变率张量的方案或基于粘度的拉普拉斯算子的方案。

人工粘度的优势是简单便捷，但是难以实现复杂的流体现象，如绳卷、屈曲等。

基于应变率张量的方案采用的 N-S 方程形式如下：

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho} \nabla p + \frac{1}{\rho} \nabla \cdot \boldsymbol{\tau} + \mathbf{f}_{ext} \quad (2-17)$$

其中， $\boldsymbol{\tau} = \mu(\dot{\gamma})\mathbf{D}$ 为剪切力，通过应变率张量 $\mathbf{D} = \nabla \mathbf{v} + (\nabla \mathbf{v})^T$ 计算，剪切率 $\dot{\gamma} = \sqrt{\frac{1}{2} \cdot \text{trace}(\mathbf{D})^2}$ 。这种方案需要有无散速度场的前提，并且容易受到粒子缺失问题的影响。

粘度计算有三种方式：人工粘度、基于应变率张量的形式以及基于拉普拉斯算子的形式。

人工粘度的计算公式如下：

$$\hat{\mathbf{v}}_i = \mathbf{v}_i + \alpha \sum_j \frac{m_j}{\rho_j} (\mathbf{v}_j - \mathbf{v}_i) W_{ij} \quad (2-18)$$

其中, α 是一个介于 0 到 1 的常数, 用于直接控制速度的增量以模拟粘滞力阻碍运动的效果。该方法一般在低粘度的流体是使用, 如水, 且计算简单。但是其本身是非物理的处理方法, 因此不能保留更复杂的粘性流的特征。

基于应变率张量的形式也被广泛使用在显式或隐式方案中。其计算公式为:

$$\mathbf{f}_{vis} = \mu(\nabla \mathbf{v} + (\nabla \mathbf{v})^T) \quad (2-19)$$

它直接根据速度梯度进行计算。但是该方法需要额外注意邻域粒子缺失问题。其中, 速度梯度的计算如下:

$$\nabla \mathbf{v}_i = \sum_j \frac{m_j}{\rho_j} (\mathbf{v}_j - \mathbf{v}_i) \nabla W_{ij} \quad (2-20)$$

基于粘度的拉普拉斯算子的方案在计算时需要离散化拉普拉斯算子, 目前有多种方案。若使用标准 SPH 离散化对粒子缺失问题非常敏感^[29], 也会在处理核函数二阶导时变号导致数值错误; 若使用两次一阶 SPH 离散化会导致引入额外的平滑^[30]。最先进的离散化方案采用有限差分结合 SPH 离散化的方法^[13], 其形式如下:

$$\nabla^2 \mathbf{v}_i = 2(d+2) \sum_j \frac{m_j}{\rho_j} \frac{\mathbf{v}_{ij} \cdot \mathbf{x}_{ij}}{|\mathbf{x}_{ij}|^2 + 0.01h^2} \nabla W_{ij} \quad (2-21)$$

该方法的好处是该量是伽利略不变量, 并且能够保持角动量守恒。

2.2.5 表面张力

表面张力是流体表面分子间相互吸引的力的结果。这种力使得流体表面凭空形成一种类似于弹性薄膜的结构, 使得流体力图减小自己的表面积。每种流体都有其特定的表面张力, 这个值受温度和其他环境因素的影响。在 SPH 方法中, 表面张力的处理也非常重要。一种常见的方法是计算颗粒的颜色梯度和曲率, 然后基于这些信息来模拟表面张力的效应。曲面法矢量场的散度是一种反映曲面曲率的方法, 其计算如下:

$$\kappa = \nabla \cdot \hat{\mathbf{n}}e$$

表面张力大小和局部曲率之间为正比关系, 其方向从流体表面指向流体内部, 计算如下:

$$f_s = \begin{cases} -\alpha\kappa\hat{\mathbf{n}}, & \text{if } ||\mathbf{s}|| \geq \tau_s \\ 0, & \text{其他} \end{cases}$$

其中, α 为正比例常数, τ_s 为限定值, 这两个值可以直接影响表面张力的模拟效果。图 2-4 呈现了流体表面张力的作用效果。

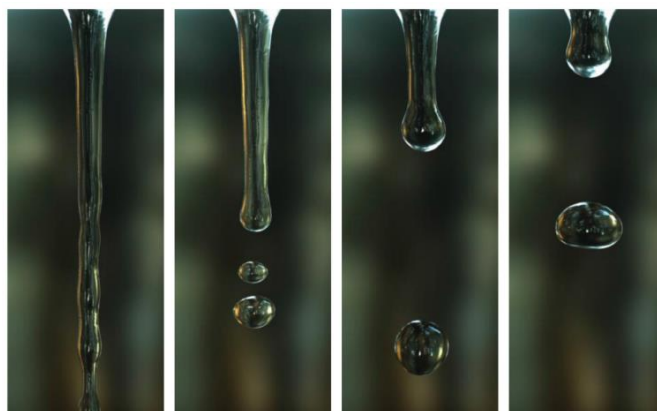


图 2-4 流体的表面张力作用, 引自文献^[31]

2.2.6 边界处理

在实际场景中, 流体与其他材料的耦合是非常常见的, 比如船在水面上航行, 沙砾在水中漂浮等。因此, 单纯模拟流体的引擎并没有什么实际的应用价值, 为此需构建流体与不同材料的耦合方法, 使 SPH 方法能够实现复杂场景的模拟。

在 SPH 方法中, 临近边界的流体粒子一般会出现邻域粒子缺失的问题, 这将导致密度漂移及压力误差等问题。同时, 若边界表面不规则, 则会得到非光滑的表面法线。因此, 非穿透力不能完全垂直于边界表面, 可能产生不必要的额外摩擦力误差。目前, 主要有两类方法用于表示边界形状和执行边界条件:

- 直接处理固体: 该方法离散化地采样固体边界, 并在其粒子之间施加约束以保持边界的形状。如此, 便能够统一地处理流体粒子和固体粒子。
- 处理流-固交互界面: 该方法根据固体边界的形状来计算交互边界处对 SPH 流体中各物理量的贡献。

这两类方法各有优缺点, 适用于不同的流-固耦合问题。通过适当处理边界条件, 可以提高 SPH 模拟的准确性和稳定性。

基于固体离散粒子的方法最为常见。其利用一组配备核函数的粒子采样固体边界, 这些粒子将对与之交互的流体粒子施加斥力, 防止流体穿透固体边界。Becker 等人^[32] 提出了一种预估-矫正方法, 但边界附近存在粒子堆叠

伪影问题。Solenthaler 等人^[33] 将边界粒子直接视为流体粒子，用于计算密度和压力，改善了粒子缺失问题，但需要小时间步长来确保不发生穿透。Ihmsen 等人^[34] 结合了 Becker 和 Solenthaler 的方法，获得了在较大时间步长下光滑的粒子分布。Akinci 等人^[35] 进一步改进了方法，实现了流体与刚体的双向耦合，同时引入了归一化的伪质量来解决边界粒子分布不均匀的问题。He 等人^[36] 引入了存储应力的交错粒子，便于准确施加边界条件，简化了流-固碰撞的处理流程。

基于流-固界面的方法通常会隐式地表达边界的几何形状。Harada 等人^[37] 提出了一种基于有向距离场（Signed Distance Fields，简称 SDF）的边界处理方法，该方法根据流体粒子到固体平板表面的距离来计算对密度贡献，从而将流体密度扩展到固体边界当中。Koschier 和 Bender^[38] 提出预先计算刚性边界的密度贡献，并将其存储在密度图中，在迭代时查询密度图以获取准确的密度值。

本课题更适用基于粒子的方法。固体边界附近的粒子分布如图 2-5 所示。

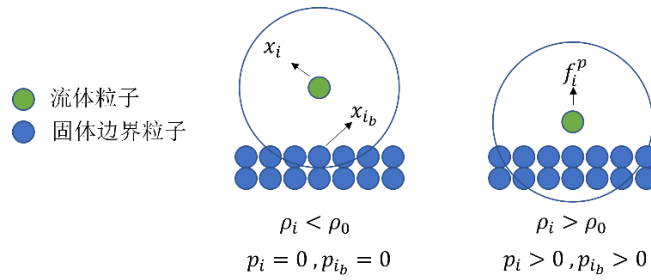


图 2-5 固体边界附近的粒子分布

图中 x_i 表示流体粒子， x_{ib} 表示边界粒子， f_i^p 表示边界粒子对流体粒子的压强贡献。当流体粒子移动到边界处时，其密度会增加。如果流体粒子过于靠近边界，则其密度会大于静止密度，这会导致压强为正，推动流体粒子远离边界。

将固体边界建模为多层粒子可以避免对固体边界采样时的粒子缺失问题，为流体粒子 x_i 采样，其密度由邻域内的流体粒子 x_{if} 和固体边界粒子 x_{ib} 共同加权求得：

$$\rho_i = \sum_{i_f} m_{i_f} W_{i_f} + \sum_{i_b} m_{i_b} W_{i_b} \quad (2-22)$$

所有的物体都使用粒子来表示，并拥有相同的静态密度和质量。至此，已经完成了对流体密度的重建，并根据该密度计算压力。

只有处于固体边界处的流体粒子才会计算固体边界的贡献。因此，需要在每一步迭代之前确认应与边界发生碰撞的粒子。由于固体边界粒子与静止流体粒子具有相同的属性，因此设定当 $\|\mathbf{x}_i - \mathbf{x}_{i_b}\| < 0.5h$ 时，视为流体粒子应与边界发生了碰撞。为了矫正该位移，Becker 等人^[32] 提出直接将粒子的位置矫正到 \mathbf{x}_{i_b} 处，但流体粒子可能穿透多个固体边界粒子，因此 zhang 等人^[39] 采用加权函数来计算边界穿透深度和方向，再进行矫正，该方法设定固体边界法向量 \mathbf{n}_i^{avg} 为流体粒子 i 穿透的所有固体边界粒子 i_b 的边界法向量 \mathbf{n}_{i_b} 的加权平均值：

$$\mathbf{n}_i^{avg} = \sum_{i_b} w_{i_b}^{avg} \mathbf{n}_{i_b} \quad (2-23)$$

$$w_{i_b}^{avg} = \max\left(0, \frac{0.5h - \|\mathbf{x}_i^*(t + \Delta t) - \mathbf{x}_{i_b}\|}{0.5h}\right) \quad (2-24)$$

根据法向量 \mathbf{n}_i^{avg} ，可将流体粒子 i 的位置矫正为：

$$\begin{aligned} \mathbf{x}_i^{**}(t + \Delta t) = & \mathbf{x}_i^*(t + \Delta t) + \\ & \frac{1}{\sum_{i_b} w_{i_b}^{avg}} \sum_{i_b} w_{i_b}^{avg} \left(0.5h - \|\mathbf{x}_i^*(t + \Delta t) - \mathbf{x}_{i_b}\|\right) \frac{\mathbf{n}_i^{avg}}{\|\mathbf{n}_i^{avg}\|} \end{aligned} \quad (2-25)$$

2.3 概念小结

欧拉模拟和拉格朗日模拟是当前主要的物理模拟方法，其中欧拉视角基于网格，而拉格朗日视角基于粒子。这两种方法各有优劣。欧拉法更擅长连续空间的时变模拟，而拉格朗日法则更擅长离散空间的处理。

SPH 方法属于拉格朗日方法，基于 SPH 的流体模拟的基本思想是将连续的流体介质离散化为 SPH 粒子，通过计算粒子的时空、物理属性来模拟流体的运动。SPH 方法是一个系统方法，其包含 SPH 离散化、邻居搜索、不可压缩求解器以及粘性求解器等组件。

邻居搜索是 SPH 系统中的基础设施之一，通常采用基于背景网格的方法来索引分布在三维空间中的粒子。

不可压缩求解器也是当前 SPH 方法的研究重点之一，现有 WSPH、PCISPH、IISPH、DFSPH 等多种压力求解器，其中 WSPH 基于状态方程，而 PCISPH、IISPH 和 DFSPH 基于预估-矫正方案，这种求解器往往具有更好的仿真稳定性和数值准确性。

粘性求解器通常有三种解决方法：人工粘度、基于应变率张量、基于拉普拉斯算子。人工粘度是非物理的简单方案，其能够快速模拟低粘度流体的粘性特征，但不能很好的保留更复杂的粘性流体行为，如绳卷、屈曲等，且这种方法无法模拟高粘度流体。基于应变率张量的方法也有许多研究成果，其中各种隐式方法能够表现出优良的性能，但是这些方案都有一个缺陷，即处于流体自由表面的粒子对邻居粒子缺失问题很敏感，容易产生视觉伪影等问题。一般采用虚粒子等方法处理流-气边界，如图 2-5 所示。基于拉普拉斯算子的方法比较直观且容易推导，处理拉普拉斯算子的手段一般有两种：标准 SPH 离散化和 SPH 离散化加有限差分。后者是更为先进的方法，该方法是维度无关的，且能够保持动量守恒，该方法依赖无散条件，因此在使用时需对速度场进行无散约束。

表面张力作用在流体表面，使流体表面总是趋向于保持最小曲率，其可以基于曲面法矢量场的散度进行计算。

边界处理中，目前多采用虚粒子等方法处理流-气边界，而处理流-固耦合，则需要考虑到粒子穿透、位置矫正等问题。目前，采用加权函数计算穿透深度和方向，再矫正位置的方案能够保证在任意形状的固体边界上，较为稳定地进行流-固耦合。

3 基于 Cross 模型的非牛顿流体仿真

3.1 剪切变稀与剪切增稠型流体

非牛顿流体（Non-Newtonian fluid）是一种流体力学的概念，其剪应力（主要是剪切力）与剪应变的关系如图 3-1 所示。剪切力可以理解为流体的粘滞力，剪切率是流体形变的速率，跟速度梯度相关。

非牛顿流体又分为纯粘性流体和粘弹性流体：

1. 纯粘性非牛顿流体
 - a) 非时变性：剪切变稀型、剪切增稠型
 - b) 时变性：震凝性流体、触变性流体等（不属于广义牛顿流体）
2. 粘弹性非牛顿流体
 - a) 宾汉塑性流体

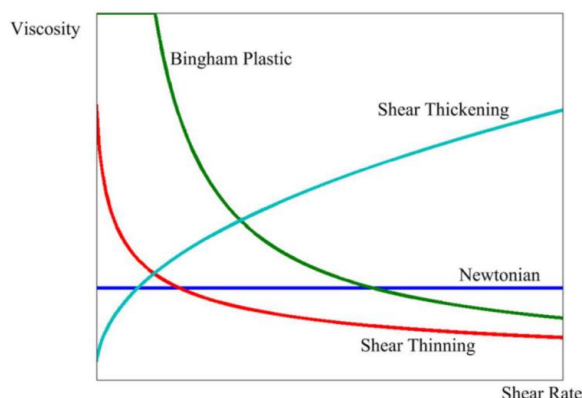


图 3-1 不同类型的非牛顿流体其剪切率与粘度的关系曲线

剪切变稀型流体，其粘度随着剪切率的增加而降低，例如当流体块落在地上的一瞬间，接触地面的流体因为速度突然改变，其粘度会降低，随后会随着流体速度的稳定而变粘。常见的剪切变稀型流体有巧克力，油漆等，如图 3-2 所示，它们的变稀行为表现为搅拌巧克力酱会使其变软；将油漆滴在桌上后，其会保持形状，但当用刷子划过时，油漆会摊开。



图 3-2 常见的剪切变稀型流体

剪切增稠型流体，其粘度随着剪切率的增加而增大，例如子弹射向液态防弹衣时，其内部的剪切增稠液的粘度在接触子弹的部位急剧升高，表现出变硬的现象，随后粘度下降。常见的剪切增稠型流体有淀粉水，如图 3-3 所示，其增稠行为表现为当手用力砸向水面时，水面急剧变硬从而阻碍手的运动，当手缓缓插入水中时，阻力却很小。



图 3-3 手与淀粉水交互

3.2 广义牛顿流体

广义牛顿流体（Generalized Newtonian Fluid）是一种用于描述非牛顿流体特性的工具。广义牛顿流体是一种理想流体，其剪应力为特定时间点的关于剪切速率的函数，与变形过程无关。广义牛顿流体满足以下流变方程，即本构方程，如式 3-1 所示：

$$\tau = \mu(\dot{\gamma})\dot{\gamma} \quad (3-1)$$

其中 τ 为剪切应力， $\dot{\gamma}$ 为剪切应变速率， μ 为运动粘度，是关于剪切应变速率的函数。用于描述粘度与剪切率之间关系的函数通常有 Cross 模型和 Carreau 模型，它们的计算公式分别如式 3-2 和式 3-3 所示：

$$\mu(\dot{\gamma}) = \mu_{\infty} + \frac{\mu_0 - \mu_{\infty}}{1 + (K\dot{\gamma})^n} \quad (3-2)$$

$$\mu(\dot{\gamma}) = \mu_{\infty} + (\mu_0 - \mu_{\infty})(1 + (K\dot{\gamma})^a)^{\frac{n-1}{a}} \quad (3-3)$$

式 3-2 和式 3-2 中， μ_0 、 μ_{∞} 表示零剪切率和无限大剪切率时的粘度； K 是用于缩放剪切率的人工参数； n 用于控制流体类型及其变化速率，式 3-2 中的 a 通常为正参数。在 Cross 模型中，当 $K = 0$ 或者 $n = 0$ 时，可以表达粘度为 μ_0 的牛顿流体； K 非零时，当 $n > 0$ ，可以描述剪切变稀型非牛顿流体，当 $n < 0$ ，可以描述剪切增稠型非牛顿流体。在 Carreau 模型中，当 $K = 0$ 或者 $n =$

1时，可以表达粘度为 μ_0 的牛顿流体； K 非零时，当 $n < 1$ ，可以描述剪切变稀型非牛顿流体，当 $n > 1$ ，可以描述剪切增稠型非牛顿流体。

此外，两种模型对不同类型的非牛顿流体的表达能力不同。设置两种模型的参数如表 3-1。

表 3-1 Cross、Carreau 参数设置

模型	μ_0	m^2/s	μ_∞	m^2/s	n (变稀/增稠)	K	a
Cross	7		1		3/-3	1	/
Carreau	7		1		3/-3	1	2

两种模型的表达能力如图 3-4 所示。横轴表示剪切率，纵轴表示粘度。可以看出，两种模型的变稀能力相当，但是增稠能力，Carreau 强于 Cross，但考虑到 Carreau 的增稠是指数型并且没有上界，这会造成数值计算不稳定的问题。因此本文选择更加稳定的 Cross 作为最终的非牛顿流体粘度模型。

此外，还有幂率流体模型、二阶流体模型、HBP 模型等。以上两种是比较常见的。Andrade 等人^[40] 和 Liu 等人^[15] 均使用了 Cross 模型；Zhu 等人^[41] 则在余维数法的框架下采用了 Carreau 模型。

对于 N-S 方程中粘度项的计算形式，本课题选用基于拉普拉斯算子的形式，该形式的优点是简单直观，且基于 SPH 离散化和有限差分的近似计算能够保持线动量及角动量的守恒。

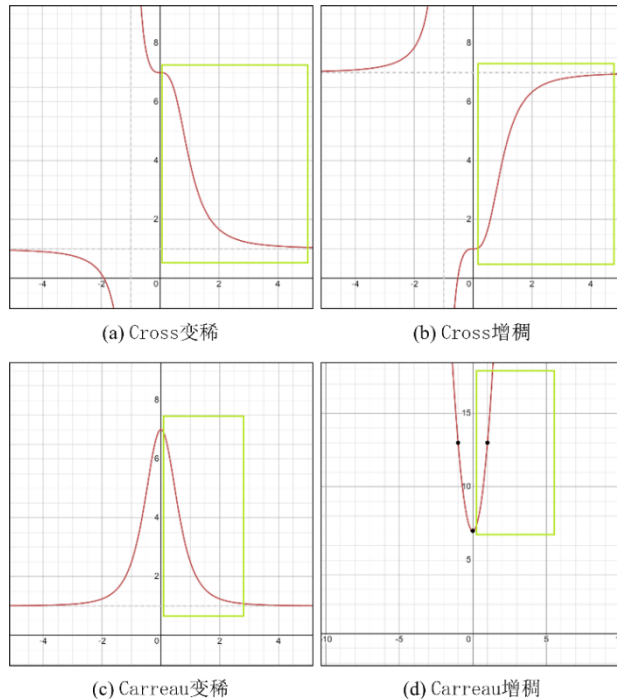


图 3-4 Cross 和 Carreau 变稀与增稠能力对比

3.3 算法实现

基于 Cross 的非牛顿流体粘度模型需要嵌入 N-S 方程的粘度项计算中。本课题选用基于拉普拉斯算子的形式,该形式的优点是简单直观,且基于 SPH 离散化和有限差分的近似计算能够保持线动量及角动量的守恒,公式如 3-4 所示:

$$\mathbf{f}_{vis} = \mu \nabla^2 \mathbf{v} \quad (3-4)$$

其中 μ 表示流体的动态粘度, \mathbf{v} 表示流体的速度。

对于 $\nabla^2 \mathbf{v}$ 部分的求解,一般有以下离散化处理方案:

1. 直接 SPH 离散化。形式为 $\nabla^2 \mathbf{v}_i = \sum_j m_j \frac{1}{\rho_j} \mathbf{v}_j \nabla^2 W_{ij}$ 。该形式对流体表面的粒子缺失问题非常敏感,会产生仿真伪影和密度漂移等问题。
2. 一次标准 SPH 离散化加有限差分。形式为 $\nabla^2 \mathbf{v}_i = 2(d+2) \sum_j \frac{m_j}{\rho_j} \frac{\mathbf{v}_{ij} \cdot \mathbf{x}_{ij}}{||\mathbf{x}_{ij}||^2 + 0.01h^2} \nabla W_{ij}$ 。该形式的好处是,其是伽利略不变量,且能够保证线动量和角动量的守恒。

因此,本课题采用方案 2 计算粘度项,完整的形式如式 3-5 所示:

$$\mu_i \nabla^2 \mathbf{v}_i = 2(d+2) \mu_i \sum_j \frac{m_j}{\rho_j} \frac{\mathbf{v}_{ij} \cdot \mathbf{x}_{ij}}{||\mathbf{x}_{ij}||^2 + 0.01h^2} \nabla W_{ij} \quad (3-5)$$

SPH 方法中,连续的流体被离散化为 SPH 粒子,式 3-4 中的动态粘度 μ 表示流体的动态粘度,但由于 SPH 方法计算的特殊性,其用来表示每个 SPH 粒子的动态粘度,如式 3-5 所示。因此,使用 Cross 模型建模非牛顿流体的算法流程如表 3-2 所示。

表 3-2 SPH 方法与 Cross 模型

算法: SPH 中嵌入 Cross 模型	
01:	while run_sim do:
02:	邻居搜索
03:	计算时间步长 Δt
04:	for 每个粒子 i do:
05:	求解外力项获得 \mathbf{v}^1
06:	根据式 3-2 更新粒子的局部粘度 μ_i^{Cross}

07:	根据式 3-5 求粘性项获得 v^2
08:	求解压力项获得 v^{t+1}
09:	for 每个粒子 i do:
10:	时间积分更新粒子位置

4 基于温度-粘度模型的非牛顿流体仿真

4.1 基于热传导方程的温度-粘度模型

流体的粘度与温度之间存在着密切的联系，对于多数流体来说，粘度会随着温度的变化而变化。对于液体，通常情况下，其粘度会随着温度的升高而减小，呈负相关关系，这是因为液体内部的分子间作用力随着温度升高而减小，分子间的距离增大，流动阻力下降，所以粘度下降。典型的例子如机油、蜂蜜等，在低温下会显得非常粘稠，难以流动，而在高温下会拥有更好的流动性。气体则相反，其粘度与温度呈正相关关系，这是因为气体的粘度主要来自于分子的碰撞，温度升高，分子的运动速度和运动频率增大，分子间的碰撞更加频繁，从而导致粘度增大。本文主要考虑液体下的情景。

在考虑流体的粘度随温度变化之前，首先需要确定热量的传递。本文主要考虑了两个传热过程，一是与环境的热交换，二是物体内部的热传递。

在大多数场景下，物体会直接暴露在空气等流体中，那么热量就会通过流体的运动从物体传递到流体，或从流体传递到物体。在许多情况下，这个过程可以用牛顿冷却定律来描述。牛顿冷却定律是物理学中描述物体冷却过程的一个经验定律，它描述的是热对流的现象。这个定律认为，物体的冷却速率（即热量传递的速率）与物体和其周围环境的温度差成正比。也就是说，温度差越大，物体失去热量的速度就越快。

牛顿冷却定律可以表示为以下形式的偏微分方程：

$$\frac{\partial T}{\partial t} = -h(T - T^{room}) \quad (4-1)$$

其中， h 表示冷却系数， T 为物体的温度， T^{room} 为环境温度。

通常，物体内部的热传递过程由热传导方程来描述。热传导方程也被称为傅里叶定律，是描述热能如何在物质中传播的物理定律。热传导方程根据物质的性质和温度梯度来描述热流的方向和速度。其基本的一维形式为： $q = -k\nabla T$ 。其中， q 表示热量， k 为热传递系数， ∇T 为温度梯度。

这个公式表明，热量会从高温区域流向低温区域，流动的速度与温度梯度成正比，与物质的热导率也成正比。将热传导方程和热能守恒定律结合起来，就可以得到描述热扩散的偏微分方程，也被称为热扩散方程： $\rho c \frac{\partial T}{\partial t} = -\nabla \cdot (k\nabla T) + Q$ 。其中， ρ 表示物质密度， c 为比热容， Q 是单位体积的热源项，表示每单位时间、单位体积内产生的热量。

本文中，不考虑外部热源项，因此 $Q = 0$ 。

$$\rho c \frac{\partial T}{\partial t} = -\nabla \cdot (k \nabla T) \quad (4-2)$$

根据式 4-1 和式 4-2，可以得到本文的温度模型：

$$\frac{\partial T}{\partial t} = \frac{\partial T^r}{\partial t^N} + \frac{\partial T^t}{\partial t} \quad (4-3)$$

其中， $\frac{\partial T^r}{\partial t} = \frac{T^{room}-T}{t^N}$ ，式中 t^N 表示物体由当前温度趋向室温所用的时间，这一项表示物体与环境的热交换。为了简化计算，该时间 t^N 通常设置为 500~1000，即迭代 500~1000 次后，物体温度趋向室温。 $\frac{\partial T^t}{\partial t} = -\frac{k}{\rho c} \nabla^2 T$ 表示物体内部的热传导，而该项就是热传导方程，其描述了场的扩散，其中 k 表示热传导系数， c 表示物体的比热容，整个系数用于控制热传导的速率。

在考虑了温度变化后，需要构建温度相关的粘度模型。本文将牛顿流体中的雷诺指数模型作为原型，如式 4-4 所示：

$$\mu = \mu^0 e^{\alpha(T^0-T)} \quad (4-4)$$

其中， μ^0 是流体在温度为 T^0 时的粘度。基于该形式，进一步向非牛顿流体做了拓展，考虑到该粘度模型是对非牛顿流体粘度的再加工，因此在 Cross 模型的基础上，推导公式如下：

$$\mu = \mu^{Cross} \frac{\mu^0 e^{\alpha(T^0-T)}}{\mu^0} = \mu^{Cross} e^{\alpha(T^0-T)} \quad (4-5)$$

4.2 流体自由表面检测

在温度变化模型中，与环境的热交换只发生在流体的自由表面粒子上，而物体内部的热传导则作用在所有粒子上。因此，在计算与环境的热交换项之前，需要确定当前时刻流体的自由表面粒子。

通常，流体的表面可以通过颜色场的梯度来区分。颜色场是一个在每个粒子位置处的值为 1、所有粒子的支撑域之外的值为 0 的标量场，其公式如式 4-6 所示：

$$c_i = \sum_j m_j \frac{1}{\rho_j} W_{ij} \quad (4-6)$$

颜色场的梯度是一个矢量场，如式 4-7 所示：

$$s_i = \sum_j m_j \frac{1}{\rho_j} \nabla W_{ij} \quad (4-7)$$

s 的模长在临近表面的地方很大，在内部很小，其方向从流体表面指向流

体内部。因此，可以通过简单的阈值判断来定位流体表面，需要注意的是，阈值在不同的流体参数下是不同的。

4.3 算法实现

温度-粘度模型用于根据当前温度条件和流体的其他属性来调整流体的粘度，使流体粘度受温度影响。在 Cross 非牛顿流体的基础上，拓展了温度-粘度模型的 SPH 算法如表 4-1 所示：

表 4-1 SPH 方法、Cross 模型、温度-粘度模型

算法：SPH 中嵌入 Cross 模型和温度-粘度模型	
01:	while run_sim do:
02:	邻居搜索
03:	计算时间步长 Δt
04:	根据式 4-5 确定流体的粒子表面
05:	for 每个粒子 i do:
06:	求解外力项获得 \boldsymbol{v}^1
07:	根据式 3-2 更新粒子的局部粘度 μ_i^{Cross}
08:	根据式 4-1 计算温度变化率并更新粒子温度
09:	根据式 4-3 调整粒子的局部粘度 μ_i^{final}
10:	根据式 3-5 求粘性项获得 \boldsymbol{v}^2
11:	求解压力项获得 \boldsymbol{v}^{t+1}
12:	for 每个粒子 i do:
13:	时间积分更新粒子位置

5 仿真验证与优化

5.1 基于 SESP 的算法验证

在 Cross 模型章节和温度-粘度模型章节中,本文构建了基于 SPH 的粘度求解器,但是并未构建完整的 SPH 计算。为了搭建完整的 SPH 框架,还需要选择合适的压力求解器。

SPH 方法种类繁多,不同的 SPH 方法间的区别也主要在于如何处理压力和密度,在 2.2.3 节中也介绍了当前的各种 SPH 方法。本文用于验证 Cross 模型和温度-粘度模型的 SPH 方法为 WCSPH,它是一种简单直观的、基于状态方程的弱可压缩 SPH 方法,其状态方程一般选用式 5-1 的形式:

$$p_i = k_1 \left(\left(\frac{\rho_i}{\rho_0} \right)^{k_2} - 1 \right) \quad (5-1)$$

计算时一般将 k_2 设为 7, $k_1 = \frac{\rho_0 c_s^2}{k_2}$, 其中, c_s 表示声音在水中传播的速度。则压力计算公式如式 5-2 所示:

$$p_i = \frac{\rho_0 c_s^2}{7} \left(\left(\frac{\rho_i}{\rho_0} \right)^7 - 1 \right) \quad (5-2)$$

在基于状态方程的方法中,一般不考虑粒子密度大于静止密度的情况,因此其压力计算结果总是为正,因此会呈现出一定的可压缩性,但在小时间步长下,这种弱可压缩性是能够接受的。

得到每个粒子的压力和密度后,可以得到压强密度比的梯度,如式 5-3:

$$\nabla \left(\frac{p_i}{\rho_i} \right) = \frac{\rho_i \nabla p_i - p_i \nabla \rho_i}{\rho_i^2} = \frac{\nabla p_i}{\rho_i} - \frac{p_i \nabla \rho_i}{\rho_i^2} \quad (5-3)$$

可以看到,其中包含 N-S 方程中的压力项。对左侧和右侧中的密度梯度进行标准 SPH 离散化(式 2-13),再移项便可得到压力项的表达式,如式 5-4:

$$-\frac{\nabla p_i}{\rho_i} = - \sum_j m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} \right) \nabla W_{ij} \quad (5-4)$$

通过以上的推导,可以得出 WCSPH 的压力项计算,将其与非牛顿流体粘度模型整合后,仿真算法流程如表 5-1 所示:

表 5-1 WCSPH 方法与非牛顿流体粘度模型

算法：WCSPH 中嵌入非牛顿流体粘度模型

```

01: while run_sim do:
02:     邻居搜索
03:     计算时间步长  $\Delta t$ 
04:     根据式 4-5 确定流体的粒子表面
05:     for 每个粒子  $i$  do:
06:         根据式 2-18 进行密度估计求得  $\rho_i$ 
07:         for 每个粒子  $i$  do:
08:             求解外力项获得  $v^1$ 
09:             根据式 3-2 更新粒子的局部粘度  $\mu_i^{Cross}$ 
10:             根据式 4-1 计算温度变化率并更新粒子温度
11:             根据式 4-3 调整粒子的局部粘度  $\mu_i^{final}$ 
12:             根据式 3-5 求粘性项获得  $v^2$ 
13:             根据式 5-4 求压力项获得  $v^{t+1}$ 
14:         for 每个粒子  $i$  do:
15:             时间积分更新粒子位置

```

5.2 使用 DFSPH 优化仿真

为了初步验证非牛顿流体的粘度模型，本文先选择了简单的 WCSPH 作为压力求解器来完善框架，进行实验。但是由于 WCSPH 是一种弱可压缩方法，其不能完全满足不可压缩条件，并且本文的粘度求解器中采用的粘度项计算形式为拉普拉斯算子形式，它所在的不可压缩 N-S 方程，是将流体的不可压缩性应用在线性动量守恒方程中推导而来的，因此该粘度项的计算非常依赖于流体的强不可压缩性。其推导过程如下所示。

线性动量守恒律如式 5-5:

$$\rho \frac{D^2 x}{Dt^2} = \nabla \cdot \mathbf{T} + \mathbf{f}_{ext} \quad (5-5)$$

其中 \mathbf{T} 为应变张量。

对于不可压缩 N-S 方程来说，经典的守恒关系如式 5-6:

$$\mathbf{T} = -p\mathbf{I} + \mu(\nabla \mathbf{v} + (\nabla \mathbf{v})^T) \quad (5-6)$$

该式基于算子分裂，其中 p 和 μ 表示压强和动态粘度。将式 5-6 代入式 5-5 中可以得到：

$$\rho \frac{D^2 \mathbf{x}}{Dt^2} = \nabla \cdot (-p\mathbf{I} + \mu(\nabla \mathbf{v} + (\nabla \mathbf{v})^T)) + \mathbf{f}_{ext} \quad (5-7)$$

其中第一项展开得：

$$\nabla \cdot (-p\mathbf{I} + \mu(\nabla \mathbf{v} + (\nabla \mathbf{v})^T)) = -\nabla p + \mu \nabla^2 \mathbf{v} + \mu \nabla(\nabla \cdot \mathbf{v}) \quad (5-8)$$

假设流体不可压缩，可以得到速度无散的约束条件，即 $\nabla \cdot \mathbf{v} = 0$ ，将该条件先验的带入式 5-8 可得到：

$$\nabla \cdot (-p\mathbf{I} + \mu(\nabla \mathbf{v} + (\nabla \mathbf{v})^T)) = -\nabla p + \mu \nabla^2 \mathbf{v} \quad (5-9)$$

以上可知，基于拉普拉斯算子的粘度项计算非常依赖不可压缩性条件。

DFSPH 是一种改进的 SPH 方法，它相对于其他 SPH 方法具有更高的稳定性和更好的可并行能力，在基本的 SPH 方法中，无散条件是通过流体连续方程隐式地满足的。然而，在实践中，这种方法往往会导致数值散度(numerical divergence)。DFSPH 通过显式地强制无散条件来解决这个问题。DFSPH 是一种预估-矫正的方法，它不直接计算压力，而是通过计算每个粒子的刚度系数，来调整压力计算。与其他采用预估-矫正方案的求解器不同的是，DFSPH 不仅有定常密度求解器，还有无散度求解器，二者分别保证不可压缩性和速度场无散。速度场无散是流体不可压缩的一个基本约束，在以往的求解中，往往会只关注密度恒定而无法完全满足速度场无散条件，这会在数值模拟中产生累加的计算误差。

DFSPH 在概念上解决了两个压力泊松方程，一个以密度不变性作为源项，另一个以速度散度作为源项。DFSPH 不是将一种解用于位置更新，将一种解用作最终速度场，而是结合这两种解来计算粒子的最终速度。这种组合的好处是，第一个压力泊松方程求解密度不变性计算不可压缩流体状态的粒子位置，但这不一定是无散速度场，而使用第二个压力泊松方程后，会得到无散速度场。

在无散求解器中，压力计算定义为如式 5-10：

$$p_i = \frac{1}{\Delta t} \frac{D\rho}{Dt} \cdot \frac{\rho_i^2}{\left\| \sum_j m_j \nabla W_{ij} \right\|^2 + \sum_j \left\| m_j \nabla W_{ij} \right\|^2} \quad (5-10)$$

其中， $\frac{D\rho_i}{Dt}$ 是物质导数， $\frac{D\rho_i}{Dt} = \sum_j m_j (\mathbf{v}_i - \mathbf{v}_j) \cdot \nabla W_{ij}$ ，并且令 $k_i^{DFSPH} = \left\| \sum_j m_j \nabla W_{ij} \right\|^2 + \sum_j \left\| m_j \nabla W_{ij} \right\|^2$ ，其代表每个粒子的刚度系数。

DFSPH 中的无散求解器的算法流程如表 5-2 所示：

表 5-2 DFSPH 方法的无散求解器

算法：DFSPH 无散求解器

```

01: while  $\left(\frac{D\rho}{Dt}\right)^{avg} > \eta^{div}$  or  $(iter < 1)$  do:
02:   for 每个粒子  $i$  do:
03:      $\frac{D\rho_i}{Dt} = -\rho_i \nabla \cdot \mathbf{v}_i^*$ 
04:   for 每个粒子  $i$  do:
05:      $\mathbf{p}_i = \frac{1}{\Delta t} \frac{D\rho_i}{Dt} \mathbf{k}_i^{DFSPH}, \mathbf{p}_j = \frac{1}{\Delta t} \frac{D\rho_j}{Dt} \mathbf{k}_j^{DFSPH}$ 
06:      $\mathbf{v}_i^* = \mathbf{v}_i^* - \Delta t \sum_j m_j \left( \frac{\mathbf{p}_i}{\rho_i^2} + \frac{\mathbf{p}_j}{\rho_j^2} \right) \nabla W_{ij}$ 

```

每个粒子的刚度系数 k_i^{DFSPH} 在每一轮迭代中，由无散求解器和定常密度求解器共用。

定常密度求解器将在迭代求解中计算密度估计，其需要解决的压力泊松方程形式如式 5-11：

$$p_i = \frac{1}{\Delta t^2} (\rho_i^* - \rho_0) k_i^{DFSPH} \quad (5-11)$$

其中估计密度的表达式为：

$$\rho_i^* = \rho_i + \Delta t \frac{D\rho_i}{Dt} = \rho_i + \Delta t \sum_j m_j (\mathbf{v}_i - \mathbf{v}_j) \cdot \nabla W_{ij} \quad (5-12)$$

以上，定常密度求解器的算法流程如表 5-3 所示：

表 5-3 DFSPH 方法的定常密度求解器

算法：DFSPH 定常密度求解器

```

01: while  $\rho^{avg} - \rho_0 > \eta$  or  $(iter < 2)$  do:
02:   for 每个粒子  $i$  do:
03:     根据式 5-12 计算密度  $\rho_i^*$ 
04:   for 每个粒子  $i$  do:
05:      $\mathbf{p}_i = \frac{\rho_i^* - \rho_0}{\Delta t^2} \mathbf{k}_i^{DFSPH}, \mathbf{p}_j = \frac{\rho_j^* - \rho_0}{\Delta t^2} \mathbf{k}_j^{DFSPH}$ 
06:      $\mathbf{v}_i^* = \mathbf{v}_i^* - \Delta t \sum_j m_j \left( \frac{\mathbf{p}_i}{\rho_i^2} + \frac{\mathbf{p}_j}{\rho_j^2} \right) \nabla W_{ij}$ 

```

DFSPH 的核心求解器如上所示。使用 DFSPH 对本课题的算法进行优化时，考虑到在 SPH 一般求解流程中，不可压缩求解器与粘度求解器会相互干扰，其计算顺序会影响各自的属性，为了尽可能保证流体的粘性行为，本文选择在 DFSPH 的无散求解之后计算粘滞力加速度，整体的算法流程如表 5-4 所示：

表 5-4 DFSPH 方法与非牛顿流体粘度模型

算法：DFSPH 中嵌入非牛顿流体粘度模型	
01:	while run_sim do :
02:	邻居搜索
04:	求解外力项获得 v^1
05:	定常密度求解器矫正密度偏差
06:	无散求解器矫正速度场的散度误差
07:	for 每个粒子 i do :
10:	根据式 4-1 计算温度变化率并更新粒子温度
11:	根据式 4-3 调整粒子的局部粘度 μ_i^{final}
12:	根据式 3-5 求粘性项获得 v^2
13:	根据式 5-4 求压力项获得 v^{t+1}
14:	for 每个粒子 i do :
15:	时间积分更新粒子位置

与用于验证粘度模型的 WCSPH 方法相比，DFSPH 不仅拥有更高的仿真稳定性与准确性，还能增大仿真时间步长，提高仿真效率。

6 实验与分析

本节通过实验验证本章方法的有效性。算法使用 taichi 编程语言实现，运行在 11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHz CPU 和 NVIDIA GeForce RTX 3050Ti Laptop GPU 显卡的个人电脑上，taichi 提供优秀的并行计算能力，算法中的邻居粒子搜索使用空间背景网格进行哈希查找，使用 taichi 原生的 GUI 提供调试时的可视化界面。仿真调试完毕后，导出 ply 文件，并使用 Houdini 软件进行渲染得到流体动画。

6.1 CFL 条件设置

CFL (Courant-Friedrichs-Lewy) 条件是一种数值稳定性准则，主要用于解决偏微分方程 (PDE) 的数值模拟问题。CFL 条件源于 Courant、Friedrichs 和 Lewy 在 1928 年提出的有关数值解偏微分方程稳定性的理论。

CFL 条件主要表述了时间步长与空间分辨率之间的关系，以确保数值解的稳定性。对于一个典型的线性双曲型偏微分方程，CFL 条件可以表示为：

$$\Delta t \leq c \cdot \frac{\Delta x}{a} \quad (6-1)$$

其中， Δt 是时间步长， Δx 是空间分辨率， a 是问题中的最大速度， c 是 CFL 数（通常取值在 0 和 1 之间）。

CFL 条件的作用如下：

- 稳定性：遵循 CFL 条件有助于保证数值解在迭代过程中的稳定性，防止数值误差的累积导致的模拟结果发散。
- 准确性：满足 CFL 条件能巩固提高数值解的准确性，使得数值解更接近真实解。

在 SPH 方法中，时间步长由两个因子决定，其公式如式 6-2：

$$\Delta t \leq \min(\Delta t_{CFL}, \Delta t_{vis}) \quad (6-2)$$

其中 Δt_{CFL} 为 CFL 条件下的时间约束， Δt_{vis} 是粘性流中特有的因子，其受到流体的粘性约束。

$$\Delta t_{CFL} \leq c_1 \frac{h}{v_{max}} \quad (6-3)$$

式 6-3 中， c_1 为 CFL 因子（通常取值在 0 和 1 之间）， h 为粒子半径， v_{max} 为流体粒子系统中的粒子的最大速度。

$$\Delta t_{vis} \leq c_2 \frac{h^2}{8\nu_0} \quad (6-4)$$

式 6-4 中, c_2 也是 CFL 因子 (通常取值在 0 和 1 之间), h 为粒子半径, ν_0 为最小剪切率下的流体粘度系数。

在 Cross 模型中, 首先会设置流体粘度的上下界, 其表示在高剪切速率和低剪切速率流体的粘度系数。在 SPH 算法中, CFL 因子往往会极大影响仿真的效率和稳定性, 在 WCSPH 等算法中, CFL 因子往往会尽可能小以保证仿真稳定运行, 但这会极大增加仿真的时间成本, 采用 DFSPH 等基于预估-校正的方案能够允许更大的时间步长以及更准确的数值模拟。在后续实验中, 会统一将 c_1 和 c_2 设置为 c 。

6.2 Cross 模型参数对流体类型的影响

本实验通过设置不同的 Cross 参数来模拟剪切变稀型和剪切增稠型流体, 如表 6-1 所示。

表 6-1 Cross 模型的参数设置

流体类型	N	μ_0	μ_∞	κ
剪切变稀型	10	25	10	5
剪切增稠型	-10	25	10	5

上述参数中, 直接决定流体类型的参数为 N, 当 $N > 0$ 时, 为剪切变稀型, 当 $N < 0$ 时, 为剪切增稠型, 当 $N = 0$, 为牛顿流体。

首先, 使用简单流体块在相同场景中模拟两种流体的流动现象, 如图 6-1 所示。

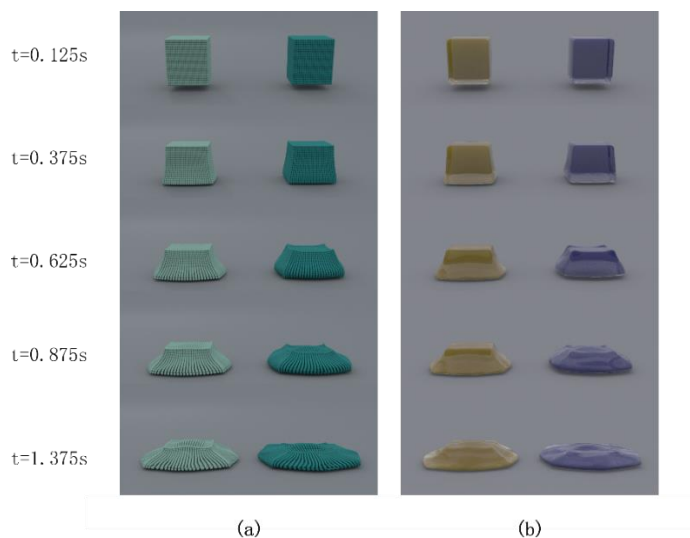
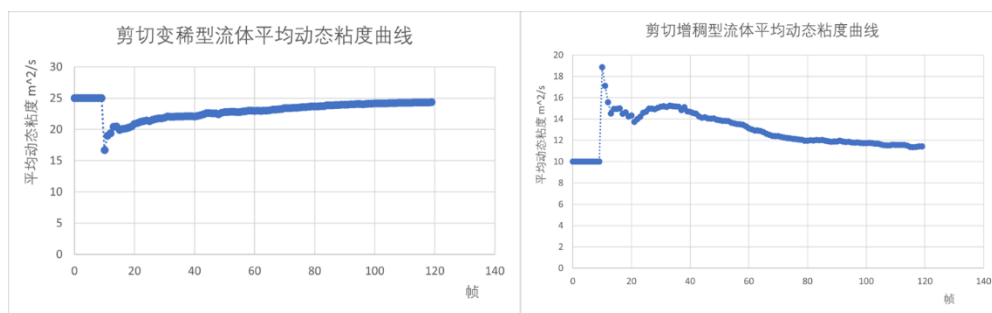


图 6-1 非牛顿流体水块实验

从图中可以看到，a 为在不同时刻的剪切变稀型流体和剪切增稠型流体的粒子模拟图，b 为其渲染图。每组图片的左侧为剪切变稀型流体，右侧为剪切增稠型流体。

在 $t=0.125s$ 时，两个流体块开始做自由落体运动；在 $t=0.375s$ 时接触地面，此时流体底部突然受击，其局部的剪切力会增大；在 $t=0.625s$ 时可以看到左侧的剪切变稀型流体底部由于剪切力的增大而导致粘度减小，局部变软，底部流体流动速度加快，而右侧的剪切增稠型流体底部的粘度则会增大，局部变硬，上下临近区域向外拱出；在 $t=0.875s$ 时，两种流体的形变特征更加明显，剪切变稀型流体的底部面积增大，而中间和顶部的流体因为其粘度较高而能保持形状，剪切增稠型流体在触地的较短时间内，高粘滞力抵消了剪应力，而后整体粘度逐渐减小，因此流动会更快；在 $t=1.375s$ 时，可以看到增稠型流体已经基本摊开，而变稀型流体还有少部分依旧保持原状。

根据上述实验，导出流体仿真期间的平均粘度，拟合成曲线后如图 6-2 所示。

图 6-2 粘度边界为 $10-25 \text{ m}^2/\text{s}$ ，流体距地面的初始高度为 5m 时的流体平均粘度曲线

流体距离地面的初始高度越高，在触地时的局部剪切率越大，粘度变化也就越大。图 6-3 表示较小高度时的流体粘度变化曲线，可以发现其更加平滑。

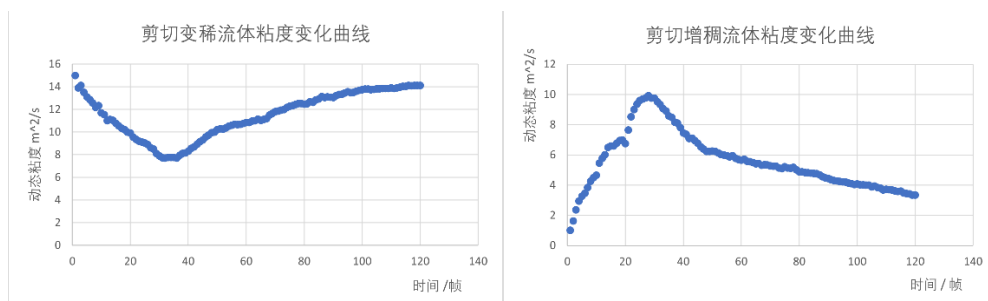


图 6-3 粘度边界为 $1-16 \text{ m}^2/\text{s}$ ，流体距地面的初始高度为 0.1m 时的流体平均粘度曲线

图 6-4 和图 6-5 为剪切变稀型和剪切增稠型流体的溃坝实验，具体内容为流体初始高度更高且在四周都有边界约束的情况下观察流体的流动现象。左侧为剪切变稀型流体，右侧为剪切增稠型流体。

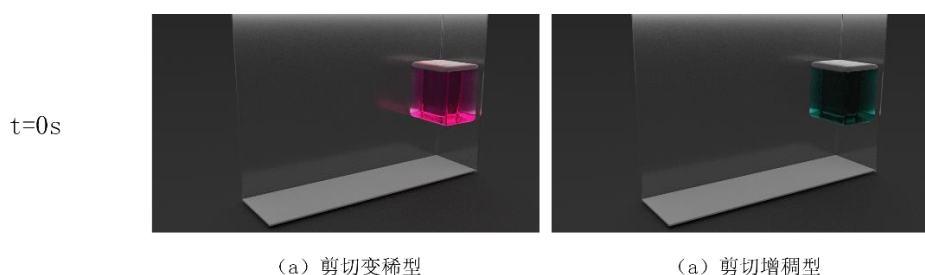
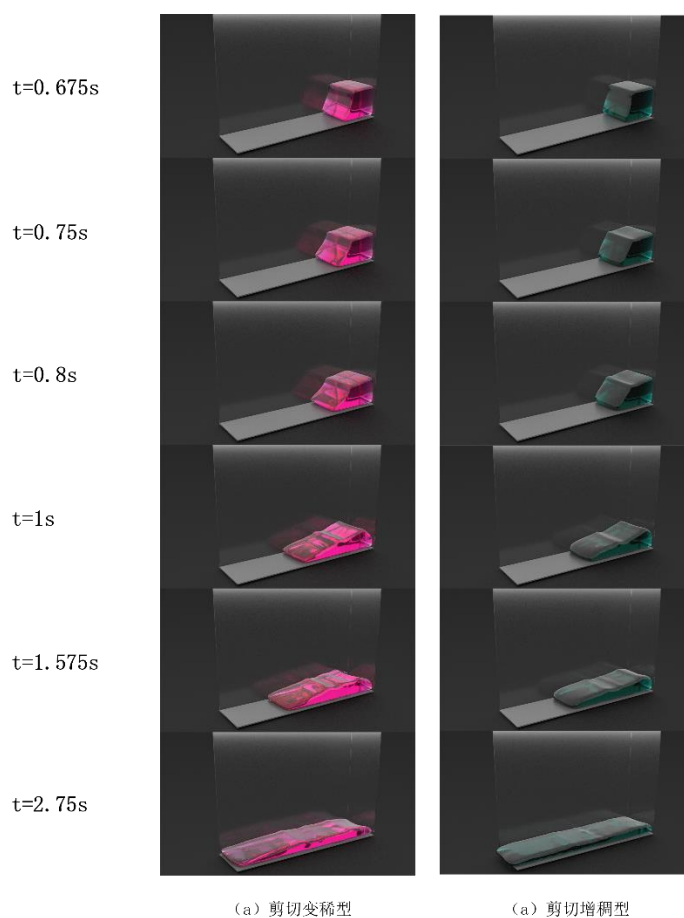


图 6-4 溃坝实验初始场景

图 6-4 表示初始场景，两流体块均离地 5m 。

**图 6-5 溃坝实验对比**

从图 6-5 可知, 0.675s~0.8s 期间, 流体跟周围边界开始交互, 由两流体的粘度特性可知, 左侧剪切变稀流体流动更快, 底部外侧区域流动更快, 右侧增稠流体还能尽量保持原状。1s~2.75s 期间, 流-固耦合处于中后期阶段, 流体趋于稳定, 因此剪切增稠型流体粘度更低, 流动更快, 且上表面更趋于平整。

流体柱与球状固体的交互场景如图 6-6 所示。该场景下流体的粘度上下界为 $15\sim30\text{ m}^2/\text{s}$ 。左侧为剪切变稀型, 右侧为剪切增稠型。

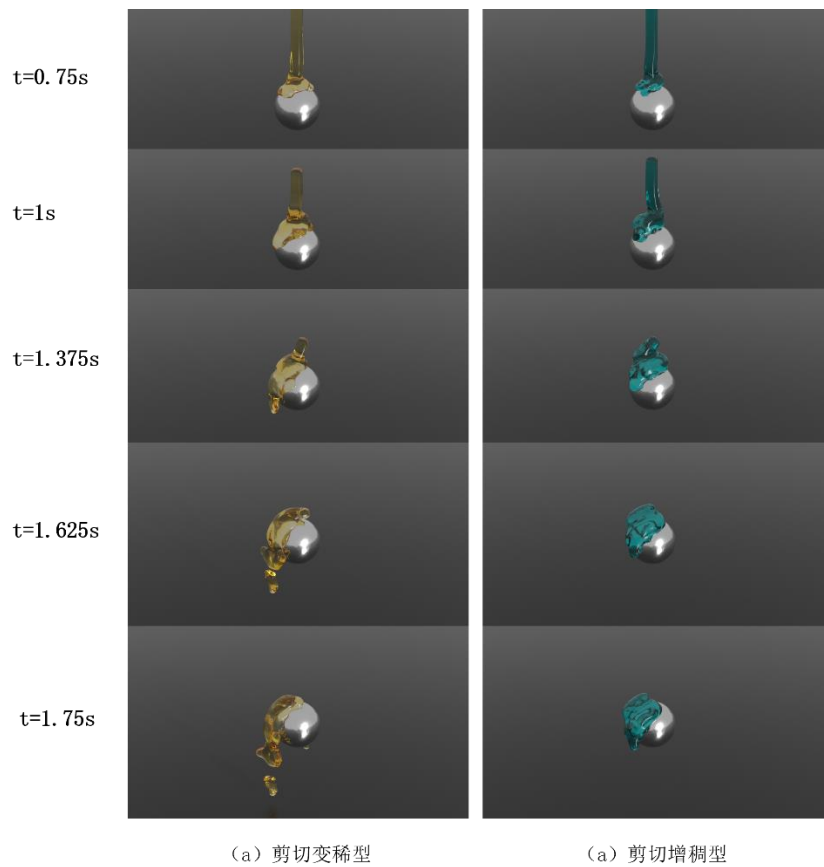


图 6-6 流体柱与球状边界交互

在该场景下，流体柱会持续地跟球形边界交互，其剪切率始终处于高水平，因此，变稀型流体会持续保持低粘度，而增稠型流体会持续保持高粘度。

6.3 温度-粘度模型实验

本章节实验在 Cross 模型构建的非牛顿流体之上，进行温度-粘度模型参数验证和模拟效果对比。

表 6-2 温度-粘度模型的参数设置

变量	描述	值
T _{room}	室温：℃	50
T ₀	流体初始温度：℃	10
α	缩放系数	0.8

表 6-2 续 温度-粘度模型的参数设置

变量	描述	值
k	热传递系数: $J/(s \cdot m \cdot K)$...
c	比热容: $J/g^{\circ}C$...

在温度-粘度模型中,主要考察热传导系数以及比热容对流动现象的影响,表 6-2 表示温度-粘度模型的参数。在本文构建的温度模型中,温度变化率由两部分组成,其中跟环境的热交换只发生在流体的表面粒子上,而流体内部的热传导则应用于所有粒子。在内部热传导项中,影响温度变化率的因子为 $\frac{k}{\rho c}$,假设密度 ρ 为常数,则热传导系数与比热容的比值会直接影响温度变化率。

图 6-6 表示三个参数设定下,流体块的流动现象。

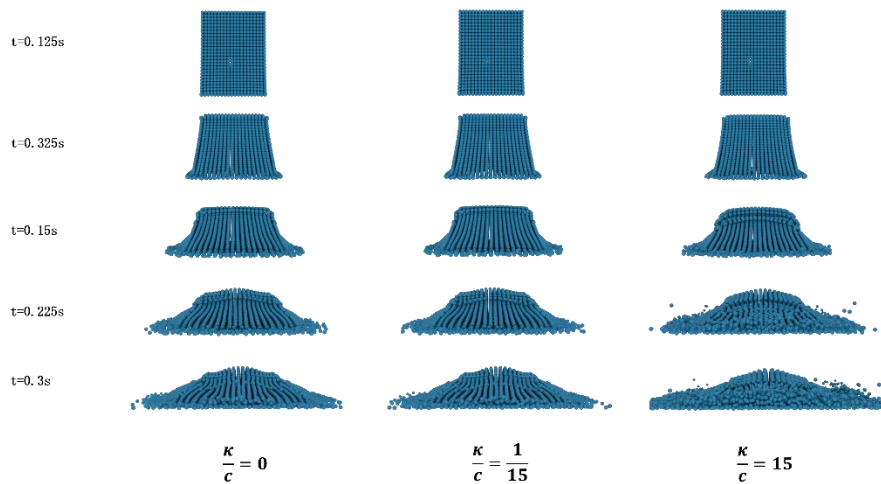


图 6-7 不同传热系数与比热容比值的对比实验

当 $\frac{k}{c}$ 不等于零时,温度-粘度模型中的两项都会参与计算,贡献温度变化率。当 $\frac{k}{c}$ 等于零时,将只有“与环境的热交换”贡献温度变化率。 $\frac{k}{c}$ 的比值越大,理论上对温度变化率源项的缩放越大,就意味着热量从流体表面传递到流体内部的速度越快,导致整体粘度减小的越快,从而使得流体块更快的融化。反之,则越慢融化。

规则流体块融化时的特征现象有顶上四角首先塌陷。这是因为在融化过程中,外界热源通常从四面八方流体块传递热量。这时,正方形的角部由于其几何形状特性,有更大的曲面区域接触到热源,所以会比中心部位接收到更多的热量,因此融化速度更快。另外,当流体块开始融化,流体的重力将使其向下移动。在正方形流体块的顶部,四角的物质在融化成流体后会因重力作用首先向下流动,形成塌陷。

从图中可知,在模拟的第一阶段($t=0.6s$), $\frac{k}{c} = 15$ 的流体块的顶上四角塌陷的更加明显,四角塌陷是一个规则物块受热融化的标志之一,其粒子图如图 6-8 所示。

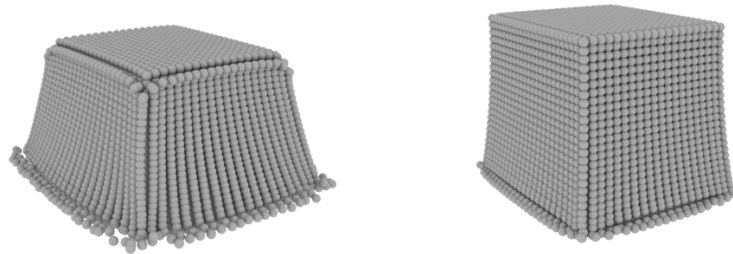


图 6-8 (左) 流体块受热后顶角塌陷 (右) 正常粘性流体模拟

在模拟的第二阶段($t=1.2s$)时,三个参数下的流体块已经与环境交互了一段时间, $\frac{k}{c} = 15$ 的流体块已经受热摊开, $\frac{k}{c} = \frac{1}{15}$ 的流体块底部周围已经散开,但是中部还有一定的流体块保持了形状,说明其粘度变化小,在时间尺度上讲则是传热慢。 $\frac{k}{c} = 0$ 的流体块则传热最慢,其形变程度比前两者都小,能够较好地保持形状。

温度无关的剪切变稀型流体和温度相关的剪切变稀型流体的对比实验如下。

表 6-3 非牛顿流体模型参数

变量	描述	值
N	用于描述流体类型的常参数	10
μ_0	最小剪切粘度: m^2/s	25
μ_∞	最大剪切粘度: m^2/s	10
κ	Cross 模型缩放系数	5

表 6-4 温度-粘度模型参数

变量	描述	值
T_room	室温: $^{\circ}C$	50
T_0	流体初始温度: $^{\circ}C$	10
α	温度-粘度模型缩放系数	0.8

表 6-4 续 温度-粘度模型参数

变量	描述	值
k	热传递系数: $\text{J}/(\text{s} \cdot \text{m} \cdot \text{K})$	2
c	比热容: $\text{J}/\text{g}^\circ\text{C}$	0.6

表 6-3 为流体模型的参数, 表 6-4 为温度-粘度模型的参数。流体仿真的粒子数量为 50k, 仿真域大小为 $3\text{m} \times 3\text{m} \times 3\text{m}$ 。实验效果如图 6-9 所示。

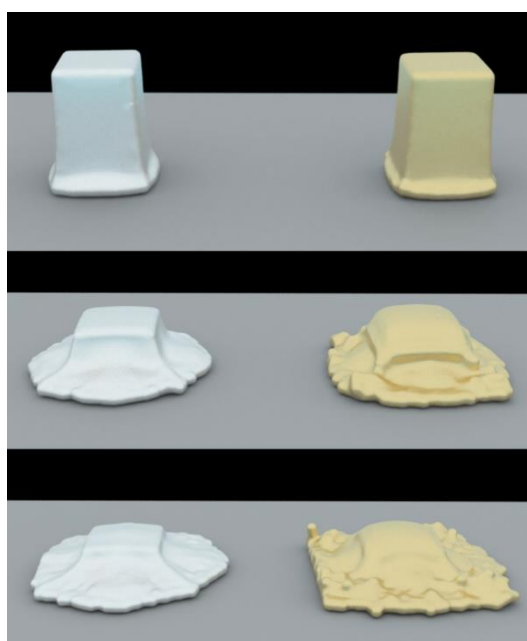


图 6-9 (左) 剪切变稀 (右) 剪切变稀与温度-粘度模型

左右两边的流体块拥有相同的本构模型, 右边的非牛顿流体施加了温度-粘度模型。以上三组图片分别表示仿真的初期、中期和后期。可以看到仿真初期时, 流体的流动现象相似, 此时右边流体的表面粒子在持续吸收热量, 并向流体内部进行热传递; 仿真中期时, 右边流体的四个角已经塌陷, 产生类似黄油块融化的现象; 到仿真后期时, 由于温度升高到了一定程度, 表面粒子的粘度已经很低, 可以看到最外层整层开始剥落。实现现象符合预期。

6.4 DFSPH 性能对比

DFSPH 用来优化仿真, 其能够保证流体的强不可压缩性。图 6-10 为 WCSSPH 与 DFSPH 在模拟非牛顿流体时, 计算粘度前的密度偏差对比 (使用粒子的密度偏差着色)。可以看出 DFSPH 的效果最好。

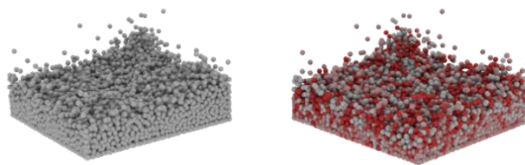


图 6-10 (左) DFSPH、(右) WCSPH 密度误差对比

使用 DFSPH 还能够显著的提高时间步长,表 6-5 表示 WCSPH 与 DFSPH 在稳定和高效模拟的情况下,不同的流体粘度所容许的最大时间步长。

表 6-5 CFL 因子对比

动态粘度 m^2/s	WCSPH	DFSPH
0.001	0.4	0.6
1	0.1	0.4
15	0.01	0.4
30	0.001	0.2

CFL 因子能直接反映出仿真的速度, CFL 因子越大,则时间步长越大,仿真越快。高粘度流体可能会增加数值计算的误差。通常, CFL 因子不会随粘度线性变化,也与粒子数量无关。从表格中可以看出,相较于 WCSPH, DFSPH 在性能方面有着显著的提升。

7 总结与展望

本文关注计算机图形学中非牛顿流体的仿真模拟问题，并实现了一种基于 SPH 方法的物理仿真方法。首先，通过构建通用的 SPH 框架并验证普通流体（如水）的模拟效果。接着，基于 Cross 模型在 SPH 框架中建模非牛顿流体（剪切变稀型和剪切增稠型），并针对该模型进行实验以验证两种非牛顿流体的仿真效果，取得了良好的视觉效果。

在 Cross 模型的基础上，嵌入了基于热传导方程的温度-粘度模型，实现了流体块与环境的热交换、自身内部的热传递，以及与温度相关的粘性流体行为（如流体块加速融化等现象）。本文先后进行了牛顿流体和非牛顿流体模型验证实验，均取得了良好的视觉效果。

高效是物理模拟的基本需求之一。完成非牛顿流体模型构建后，由于普通 SPH 框架（如 WCSPH 等）无法满足速度场无散的基本要求，本文选择使用 DFSPH 进行仿真优化。通过加速 N-S 方程中的压力项求解提高仿真速度和保证无散条件来优化算法的准确性，本文对比了两种 SPH 方法，并验证了 DFSPH 的流体模拟能力。

本文的研究内容主要体现在三个方面。第一，本文针对剪切变稀型和剪切增稠型流体的流变特征，采用 Cross 模型对其进行建模。第二，针对温度相关的粘性流体行为，向 SPH 算法中嵌入了基于热传导方程的温度-粘度模型，该模型分为温度模型和粘度模型，粘度模型没有关于非牛顿流体的描述，因此需要向非牛顿流体拓展。第三，本文采用 DFSPH 进行仿真优化，考虑到压力求解器和粘性求解器间的干扰以及 Cross 模型、温度-粘度模型的相互作用，本文改进了整个算法的流程以解决三者间的耦合问题。

此外，本文的方法还有很多提升空间。首先，对于粘度求解，本文使用的是显式积分方法，这在一定程度上限制了算法的时间步长。其次，根据对广义牛顿流体模型的分析对比，虽然本文选择的是更为稳定的 Cross 模型，但 Carreau 模型能够描述更为极端的剪切增稠情况，理论上，该模型的表达能力更强。最后，温度-粘度模型中目前只考虑了与环境的热传导和流体内部的热传导，还可以添加外部热源项来模拟物体因接触或受到其他辐射而产生的局部粘度变化，如点热源照射下，面向热源的一侧流体应更快融化等。

参考文献

- [1] Xiao Y, Chan S, Wang S, et al. An adaptive staggered-tilted grid for incompressible flow simulation[J]. ACM Transactions on Graphics (TOG), 2020, 39(6): 1-15.
- [2] Chen Y L, Meier J, Solenthaler B, et al. An extended cut-cell method for sub-grid liquids tracking with surface tension[J]. ACM Transactions on Graphics (TOG), 2020, 39(6): 1-13.
- [3] Goldade R, Wang Y, Aanjanya M, et al. An adaptive variational finite difference framework for efficient symmetric octree viscosity[J]. ACM Transactions on Graphics (TOG), 2019, 38(4): 1-14.
- [4] Macklin M, Müller M. Position based fluids[J]. ACM Transactions on Graphics (TOG), 2013, 32(4): 1-12.
- [5] Winchenbach R, Akhunov R, Kolb A. Semi-analytic boundary handling below particle resolution for smoothed particle hydrodynamics[J]. ACM Transactions on Graphics (TOG), 2020, 39(6): 1-17.
- [6] Zhai X, Hou F, Qin H, et al. Fluid simulation with adaptive staggered power particles on gpus[J]. IEEE Transactions on Visualization and Computer Graphics, 2018, 26(6): 2234-2246.
- [7] Chen Y, Li W, Fan R, et al. Gpu optimization for high-quality kinetic fluid simulation[J]. IEEE Transactions on Visualization and Computer Graphics, 2021, 28(9): 3235-3251.
- [8] Hu Y, Fang Y, Ge Z, et al. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling[J]. ACM Transactions on Graphics (TOG), 2018, 37(4): 1-14.
- [9] Nakanishi R, Nascimento F, Campos R, et al. RBF liquids: an adaptive PIC solver using RBF-FD[J]. ACM Transactions on Graphics (TOG), 2020, 39(6): 1-13.
- [10] Fang Y, Qu Z, Li M, et al. IQ-MPM: an interface quadrature material point method for non-sticky strongly two-way coupled nonlinear solids and fluids[J]. ACM Transactions on Graphics (TOG), 2020, 39(4): 51: 1-51: 16.
- [11] Schechter H, Bridson R. Ghost SPH for animating water[J]. ACM Transactions on Graphics (TOG), 2012, 31(4): 1-8.
- [12] Takahashi T, Dobashi Y, Fujishiro I, et al. Implicit formulation for SPH - based viscous fluids[C].//Computer Graphics Forum. 2015, 34(2): 493-502.

- [13] Brookshaw L. A method of calculating radiative heat diffusion in particle simulations[J]. Publications of the Astronomical Society of Australia, 1985, 6(2): 207-210.
- [14] Weiler M, Koschier D, Brand M, et al. A physically consistent implicit viscosity solver for SPH fluids[C]//Computer Graphics Forum. 2018, 37(2): 145-155.
- [15] Liu S, He X, Wang W, et al. Adapted SIMPLE Algorithm for Incompressible SPH Fluids With a Broad Range Viscosity[J]. IEEE Transactions on Visualization and Computer Graphics, 2021, 28(9): 3168-3179.
- [16] Bender J, Koschier D. Divergence-free SPH for incompressible and viscous fluids[J]. IEEE Transactions on Visualization and Computer Graphics, 2016, 23(3): 1193-1206.
- [17] Peer A, Ihmsen M, Cornelis J, et al. An implicit viscosity formulation for SPH fluids[J]. ACM Transactions on Graphics (TOG), 2015, 34(4): 1-10.
- [18] Peer A, Teschner M. Prescribed velocity gradients for highly viscous SPH fluids with vorticity diffusion[J]. IEEE transactions on visualization and computer graphics, 2016, 23(12): 2656-2662.
- [19] 张文生. 科学计算中的偏微分方程有限差分法[M]. 高等教育出版社, 2006.
- [20] 李人宪. 有限体积法基础[J]. 2008.
- [21] 龙述尧. 无网格方法及其在固体力学中的应用[M]. 科学出版社, 2014.
- [22] Monaghan J J. Smoothed particle hydrodynamics[J]. Annual review of astronomy and astrophysics, 1992, 30: 543-574.
- [23] Monaghan J J. Smoothed particle hydrodynamics[J]. Reports on progress in physics, 2005, 68(8): 1703.
- [24] Ihmsen M, Orthmann J, Solenthaler B, et al. SPH fluids in computer graphics[C]. //Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation. Aire-la-Ville, Switzerland: Eurographics Association, 2014: 21-42.
- [25] Becker M, Teschner M. Weakly compressible SPH for free surface flows[C].//Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation. Aire-la-Ville, Switzerland: Eurographics Association, 2007: 209-217.
- [26] Solenthaler B, Pajarola R. Predictive-corrective incompressible SPH[J]. ACM Transactions on Graphics, 2009, 28(3): 341-352.
- [27] Ihmsen M, Cornelis J, Solenthaler B, et al. Implicit Incompressible SPH[J]. IEEE Transactions on Visualization & Computer Graphics, 2014, 20(3): 426-

435.

- [28] Bender J, Koschier D. Divergence-free smoothed particle hydrodynamics[C]. //Proceedings of the 14th ACM SIGGRAPH/Eurographics symposium on computer animation. 2015: 147-155.
- [29] Price D J. Smoothed particle hydrodynamics and magnetohydrodynamics[J]. Journal of Computational Physics, 2012, 231(3): 759-794.
- [30] Flebbe O, Muenzel S, Herold H, et al. Smoothed particle hydrodynamics: physical viscosity and the simulation of accretion disks[J]. The Astrophysical Journal, vol. 431, no. 2, pt. 1, p. 754-760, 1994.
- [31] He X, Wang H, Zhang F, et al. Robust simulation of sparsely sampled thin features in SPH-based free surface flows[J]. ACM Transactions on Graphics (TOG), 2014, 34(1): 1-9.
- [32] Becker M, Tessendorf H, Teschner M. Direct forcing for lagrangian rigid-fluid coupling[J]. IEEE Transactions on Visualization and Computer Graphics, 2009, 15(3): 493-503.
- [33] Solenthaler B, Schläfli J, Pajarola R. A unified particle model for fluid– solid interactions[J]. Computer Animation and Virtual Worlds, 2007, 18(1): 69-82.
- [34] Ihmsen M, Akinci N, Gissler M and Teschner M. Boundary handling and adaptive time-stepping for PCISPH[C]// Proceedings of the Seventh Workshop on Virtual Reality Interactions and Physical Simulations. 2010: 79–88.
- [35] Akinci N, Ihmsen M, Akinci G, et al. Versatile rigid-fluid coupling for incompressible SPH[J]. ACM Transactions on Graphics (TOG), 2012, 31(4): 1-8.
- [36] He X, Liu N, Li S, et al. Local poisson SPH for viscous incompressible fluids[C]//Computer Graphics Forum. Blackwell Publishing Ltd, 2012, 31(6): 1948-1958.
- [37] Harada T, Koshizuka S, Kawaguchi Y. Smoothed particle hydrodynamics in complex shapes[C]//Proceedings of the 23rd Spring Conference on Computer Graphics. 2007: 191-197.
- [38] Koschier D, Bender J. Density maps for improved SPH boundary handling[C]//Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation. 2017: 1-10.
- [39] 张雅斓. 基于物理及数据驱动的非牛顿流体动画[D].北京科技大学,2020.DOI:10.26945/d.cnki.gbjku.2020.000252.
- [40] Andrade L F D S, Sandim M, Petronetto F, et al. SPH fluids for viscous jet buckling[C]//2014 27th SIBGRAPI Conference on Graphics, Patterns and Images. IEEE, 2014: 65-72.

- [41] Zhu B, Lee M, Quigley E, et al. Codimensional non-Newtonian fluids[J]. ACM Transactions on Graphics (TOG), 2015, 34(4): 1-9.

在学取得成果

一、 在学期间所获的奖励

(以下, “民考汉、内地班、区内藏族” 简称 “民族”)

2019-2020 “民族” 学生 (新生) 二等 奖学金	北京科技大学	2020.05
2019-2020 优秀学生干部	北京科技大学	2020.11
2019-2020 “民族” 学生二等奖学金	北京科技大学	2020.11
2021iCAN 全国大学生创新创业大赛北 京赛区二等奖	赛事组委会	2021.10
2020-2021 “民族” 学生三等奖学金	北京科技大学	2021.11
2020-2021 校级三好学生	三好学生	2021.11
2021 华北五省计算机应用大赛本科组 一等奖	北京市教育委员会	2021.11
2021-2022 “民族” 学生三等奖学金	北京科技大学	2022.11

二、 在学期间发表的论文

无

三、 在学期间取得的科技成果

无

致 谢

在此毕业设计完成之际，我想向所有给予我支持和鼓励的人表示衷心的感谢。

首先感谢我的指导教师张雅斓老师，她对我的研究工作给予了细致指导、宝贵建议和无私帮助。她严谨的治学态度、卓越的专业素养以及对学术追求的热情深深地影响着我，使我获益良多。

其次感谢北京科技大学 AI3D 实验室的班晓娟教授、王笑琨老师，以及组内的师兄师姐们，他们在这段时间里给予我的帮助和支持得以让我顺利完成毕业设计。

另外，我要向我本科生的导师王成耀教授，以及所有学院的教师们表达我深深的感谢。他们的专业知识深厚，行事严谨，对我产生了深远的影响。在我的本科学习期间，他们不厌其烦地帮我解答疑问，为我提供了无数宝贵的建议。

最后，我要感谢我的家人，感谢他们始终支持着我，特别是，我想感谢我的女朋友，她的爱、理解和无尽的耐心在我的工作和生活中起到了至关重要的作用。她的支持和鼓励帮助我克服了各种挑战，使我能够专注于完成这项任务。

在短暂而美好的四年时光里，北京科技大学的点滴回忆如同闪耀的星辰，永远璀璨于心。我将满怀梦想，踏上新的征程，继续书写生命中的华彩篇章。

