涉密论文 □　公开论文 □

浙 江 大 學

本 科 生 毕 业 论 文

题目　三维殷集的布尔代数算法和程序实现

姓名与学号　邱云昊 3170102133

指导教师　张庆海

年级与专业　2017级数学与应用数学

所在学院　数学科学学院

递交日期　2021/5/28

浙江大学本科生毕业论文（设计）承诺书

1. 本人郑重地承诺所呈交的毕业论文（设计），是在指导教师的指导下严格按照学校和学院有关规定完成的。

2. 本人在毕业论文（设计）中除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得**浙江大学**或其他教育机构的学位或证书而使用过的材料。

3. 与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

4. 本人承诺在毕业论文（设计）工作过程中没有伪造数据等行为。

5. 若在本毕业论文（设计）中有侵犯任何方面知识产权的行为，由本人承担相应的法律责任。

6. 本人完全了解**浙江大学**有权保留并向有关部门或机构送交本论文（设计）的复印件和磁盘，允许本论文（设计）被查阅和借阅。本人授权**浙江大学**可以将本论文（设计）的全部或部分内容编入有关数据库进行检索和传播，可以采用影印、缩印或扫描等复制手段保存、汇编本论文（设计）。

作者签名：　　导师签名：

签字日期：　　年　月　日　　签字日期　　年　月　日

致谢

首先，我要感谢大学里教过我的所有老师们，许多课程不仅让我对知识有了更深的理解，也对生活有了更深的理解。这里我特别要感谢我的导师张庆海老师，在相处的快一年中，一直耐心的教导我鼓励我，为我的科研之路提供了很多的帮助，同时老师学数学做科研的经验、知行合一的观念都让我获益匪浅。

其次，我要感谢学长学姐们，在我懵懂无知的时候给予我经验和帮助；感谢我的朋友们，他们都是各自专业的佼佼者，与他们的交流让我对不同的专业都有了较多的了解，他们优秀的学习方法和经历影响着我、鞭策着我；感谢我的室友们，四年深厚的情谊，我们一起分享了无数喜悦和汗水。

最后，我想感谢我的父母、家人们，至始至终毫无保留地支持着我，我会继续努力！

摘要

我们为三维连续介质的建模提供了一种工具——三维殷集，它可以用于多相流背景下具有复杂拓扑结构的流相的表示、界面几何大变形和流相拓扑变化的研究，也可以用于其它强调几何和拓扑结构的领域的研究。殷集是欧氏空间中边界有界的正则半解析开集，我们证明了三维殷集的边界可以唯一地分解为一系列有向黏合紧曲面，并由此在常数时间内得到殷集的连通分量个数和洞的个数。为了捕捉拓扑的变化，实现具有复杂拓扑的介质的求交和求并，我们以殷集的边界分解为基础，在殷集的边界表示空间上设计了高效的布尔运算算法，并建立殷空间与边界表示空间上布尔代数之间的同构。在具体实现时，我们的输入是一些有向黏合紧曲面的三角剖分，我们设计了相应的三角剖分算法和三角形求交算法。我们用 C++ 编程语言实现了殷集的表示和布尔运算的软件包，在多个具有复杂几何和拓扑结构的数据集上测试，都取得了正确的结果。

关键词：三维殷集，布尔代数，有向黏合紧曲面，三角剖分

Abstract

We propose a model of three-dimensional continua, namely three-dimensional Yin set, for the study of complex topology, large geometry deformations and topological changes in the context of multiphase flows. Yin sets are regular semianalytic open sets with bounded boundary. We further prove that the boundary of a three-dimensional Yin set can be uniquely decomposed into a set of oriented glued compact surfaces, from which Betti numbers can be deduced in constant time. To capture the topological changes and realize boolean operations on the continua with complex topology, we design efficient algorithms of boolean operations on the boundary representation space of Yin sets, based on the unique boundary decomposition. We further prove the isomorphism between boolean algebra of Yin space and boolean algebra of boundary representation space by the boundary-to-interior map. Since our input is the triangulation of some oriented glued compact surfaces, we design triangulation algorithm as well as triangle-intersection algorithm. We validate and verify the correctness of our algorithms and implementation of packages by a number of tests with complex geometry and non-trivial topological structures.

Key words: three-dimensional Yin set, boolean algebra, oriented glued compact surface, triangulation.

目录

第一部分　毕业论文

第二部分　毕业论文开题报告

第一部分

毕业论文

1 绪论

有物理意义的区域（在均匀连续介质意义下）是无处不在的，对这种区域的建模在很多工程和科学领域中有着基础而重要的意义。通常，对有物理意义区域的建模是一个被称为实体建模的成熟领域的研究目标。近年来多相流领域发展迅速，为了进行更深入的研究，需要能够严格分析、保持几何和拓扑结构的三维建模方法，而已有的实体建模方法并不能满足这样的要求。在本文中，我们提出了一种新的实体建模方法，即三维殷集方法，满足了上述要求。我们完善了三维殷集的理论部分，设计了布尔代数算法，并用程序实现了殷集的表示和布尔运算。下面先介绍实体建模已有的一些研究。

1.1 传统实体建模方法

三维实体建模是一组将三维有物理意义区域用于数学和计算机建模的一致的准则，它包含了聚焦“信息完全”的实体表示的一整套理论、技术和系统[1]，并能够自动计算表示实体的几何性质。它与几何建模、计算机图形学等相关领域的主要差别在于它强调物理的保真性。实体建模是一项非常吸引人的技术，它为一些本需要人来手工操作的任务，如制图、有限元分析的网格生成等，提供了自动化的工具。实体建模是三维计算机辅助设计（CAD）的基础，它支持物理对象的数字模型的创造、交换、可视化、动画制作、查询和注释。它在三维计算机图形学、计算机视觉、机器人学等学科领域同样发挥着巨大的作用。三维实体建模的运用提高了复杂零件设计的速度和灵活性，在许多行业，如医疗、娱乐等，都有着广泛的应用[2]。我们在这一节将会介绍几种传统的实体建模方法。

1.1.1 构造实体几何

构造实体几何（CSG）方法[3] 将实体定义为：基元进行一系列运算（如求交、求并）得到的结果。这里的基元是一些预先给定的参数化的实体，如正方体、球、圆柱等等。基元和它们经过运算得到的结果可以进行平移和伸缩。一个CSG 实体的描述往往是精确的，可以对它进行参数化和编辑，使得它适用于程序化、高层次的建模。但是它并不携带任何关于实体连通性甚至存在性的信息。

事实上，一个非空的 CSG 描述可以表示一个空的实体，比如对两个不相交的实体求交。为了解决这个问题，一种思路是把边界考虑进来，这引出了实体建模的另一个主流方法，边界表示法（BREP）[3]。

1.1.2 边界表示法

边界表示法（BREP）的核心思想是：一个实体的边界可以通过面的集合来表示，每一个面可以通过边的集合来表示，每一条边可以通过顶点来表示。边界表示法强调实体外表的细节，它详细记录了实体的所有几何信息和拓扑信息，将面、边、顶点分层记录，建立层与层之间的联系。为了保证 BREP 表示的实体是合法的，它的边界必须是嵌入在三维空间的可定向闭合流形[4]。BREP 中表示边界的面往往是一些多边形，被称为多边形网格。三角形是最简单的多边形，所有多边形都可以分割成若干三角形。由三角形面组成的 BREP 被称为三角网格，也就是代数拓扑中的三角剖分，它在研究中被广泛地使用，尤其是在 GPU 当中，这种数据结构上的运算非常高效。

1.1.3 其他实体建模表示方法

CSG 和 BREP 是实体建模最常用的两种方法，对于其它的方法，基于[2] 和[5]，我们在这里做一个简略的介绍。参数化基元的实例化方法预先给定一个包含多族基元的库，每一族的基元通过确定若干参数后变成实例，也即实体的表示。例如，库中的螺栓基元，通过确定它的长度、宽度等参数可以表示所有的螺栓。这种方法并不能将实例进行组合产生新的更复杂的实例。空间枚举法将建模空间划分成固定大小的网格，通常是立方体网格，每一个网格称为一个单元，会用一个点比如网格的中心来表示。实体由它所占据的空间内的所有网格构成的集合表示。单元分解法是一般化的空间枚举法，区别在于每一个单元的大小不再是固定的，它可以看作是以立方体为基元的 CSG 的一个特例。单元分解法要求不相邻的实体只能进行求并操作，从而避免了 1.1.1 中提到的非空 CSG 描述表示一个空的实体的情况。扫描法是一种适合运动规划的方法，它预先给定一个运动集合和轨迹，实体由运动集合沿轨迹扫过的体积表示。这种表示在计算两个实体相对运动时的动态干涉、机器人导航等应用背景下特别有用。

1.1.4 同调代数

同调代数是一种非常强大的工具，它可以通过对局部数据的计算得到空间的全局特征，它将代数、组合、计算和拓扑很好地结合在一起[6]。同调代数方法可以与 BREP 和 CSG 结合，例如，在 CSG 中，当基元为立方体时，它所表示的实体就是一个立方复形，就可以依此构建链复形，计算同调群，从而实体的连通分量、洞的个数都可以得到。BREP 中对边界的三角剖分可以看作是构造了一个单纯复形，也可以作类似的计算得到 Betti 数。

1.2 布尔代数和 r-sets

在建模过程中，往往需要在建模空间上定义一些操作。毕竟，建模的主要目的是为了回答关于建模对象的一些问题，而这通常涉及对建模对象的操作。从理论而言，建模空间需要与问题相适应，从计算而言，这些操作需要给出代数化、构造化的定义，从而可以将其设计成具体的算法。

布尔运算是实体建模过程中重要而有用的操作，大部分实体建模方法都会支持布尔运算。布尔运算在实体建模中的作用包括[1]：可以对一些已有的简单实体进行有限次布尔运算得到具有复杂拓扑的新的实体；能够检测空间中物体的碰撞和干扰；能够建模和模拟制造过程，比如铣削和钻孔。借助布尔运算，可以刻画拓扑变化，处理其中的退化情形。布尔运算设计的一个基本的要求是对象在布尔运算的作用下是封闭的，这对于通常集合论中的交并补是不一定成立的，必须将其正则化，与此相关的是 r-sets 的理论。r-sets[7] 是欧式空间中有界的正则半解析闭集。它的正则性捕捉到了连续介质中低维特征的缺失；半解析则保证了边界的行为良好。r-sets 精确的数学定义保证了可以在其上建立理论正确的布尔运算算法，使得 r-sets 在布尔运算下是封闭的[2]，从而为实体建模中的一些重要的表示方法如构造实体几何等奠定基础。

1.3 二维殷集

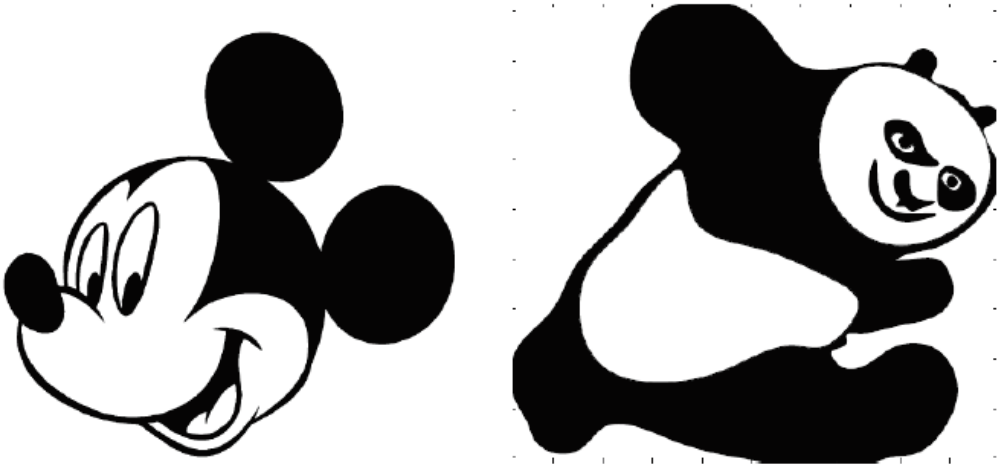
用二维殷集进行建模是最近提出的一种实体建模方法[8]。二维殷集是 R^2 中边界有界的正则半解析的开集。上述定义中，正则性保证了在建模时可以忽略那

些低维度的信息，很好地刻画了有物理意义的区域；半解析和边界有界性保证了殷集在计算机中是能够表示的，是可计算的，这些定义与 r-sets 是类似的。二维殷集与 r-sets 的区别在于，殷集要求集合是开的，开集保证了殷集边界表示的唯一性，从而能够建立殷空间上布尔代数和边界表示空间上布尔代数之间的同构。所以，殷集是连续介质的恰当数学模型。

在[8] 中，证明了二维殷集边界的拓扑结构定理，从而每一个殷集的边界可唯一地分解为若干两两几乎不交且分段解析的一维有向约当曲线。事实上，殷集可以看作是 r-sets 和 BREP 的结合与推广。殷集上定义了高效而简单的布尔运算，可以捕捉拓扑变化、处理退化情形，并且由它的边界表示可以非常自然地在常数时间内得到实体的连通分量和洞的个数。殷集是由一些可计算的数学性质定义的，独立于特定的表现形式或应用程序，可以很好地应用于多相流领域和其它需要利用几何和拓扑信息进行严格分析的领域。图 1.1 给出了二维殷集上布尔运算的一个例子。

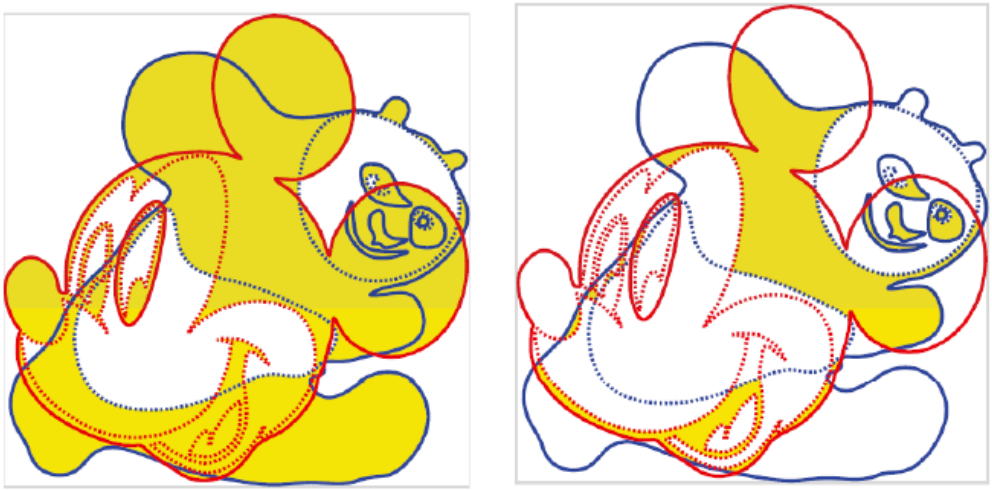
1.4 问题和动机

现有的 CSG、BREP 等三维实体建模的表示方法，或是忽略了模型的拓扑信息和几何信息，或是需要较多的内存才能计算得到拓扑信息，效率不高，这并不适用于多相流的研究。在计算连通分量个数、洞的个数等信息时，无论是基于立方复形还是单纯复形，时间复杂度都至少为 $O(n)$[6, 9]，n 为三角形或立方体的个数，因为至少需要将这些基元遍历一遍。当拓扑变化发生时，会在边界上产生非流形点，已有的布尔代数算法很难处理这些非流形点。r-sets 要求实体是闭的，这对于某些领域（如多相流）的研究是不自然的，并且会导致实体边界表示的不唯一。此外，r-sets 上建立的布尔运算算法非常复杂且效率不高。二维殷集将 r-sets 中要求的闭集条件改为开集后，则可以使边界有唯一的表示，在此基础上建立更简单高效的布尔代数算法。然而二维殷集并不能简单地推广到三维，三维的情况更加复杂，需要重新证明相关定理并设计布尔运算的算法。



(a) 米老鼠形殷集，黑色区域代表连续介质

(b) 熊猫形殷集



(c) 求并结果

(d) 求交结果

图 1.1: 二维殷集求交并的一个例子

1.5 本文贡献

本文致力于解决三维殷集的布尔代数算法和程序实现。具体贡献可分为以下三点，（1）我们完善了三维殷集的理论部分，给出了三维殷集的边界表示定理，即三维殷集的边界可唯一地分解为一些有向黏合紧曲面；（2）在边界表示定理的基础上，我们设计了边界表示空间上的布尔运算算法，建立了殷空间上布尔代数与边界表示空间上布尔代数的同构，并给出了证明和细节；（3）我们用 C++ 编程语言实现了设计的算法，并完成了一些具有复杂几何和拓扑结构的例子的求交和求并，将结果可视化后展示在论文当中。

本文的剩余部分按如下进行组织：第 2 节将会阐述三维殷集的理论部分；第 3 节会介绍我们设计的布尔运算算法，包括证明和细节；第 4 节将会展示程序的测试结果；第 5 节会总结我们已完成的工作，并提出未来还需解决的问题。

2 三维殷集理论

2.1 三维殷集定义与准备工作

在介绍殷集定义之前，我们先引入一些概念。

定义 2.1. 在拓扑空间 $\mathcal{X}$ 中，一个集合 $\mathcal{P} \subset \mathcal{X}$ 的补集为 $\mathcal{X}\backslash\mathcal{P}$，记作 $\mathcal{P}'$。$\mathcal{P}$ 的闭包为所有包含 $\mathcal{P}$ 的闭集的交，记作 $\mathcal{P}^-$。$\mathcal{P}$ 的内部为 $\mathcal{P}$ 的所有内点构成的集合，记作 $\mathcal{P}^\circ$。$\mathcal{P}$ 的外部是指 $\mathcal{X}$ 的补集的内部，记作 $\mathcal{X}^\perp := \mathcal{P}'^\circ := (\mathcal{P}')^\circ$。对于一个开集 $\mathcal{P}$，如果满足 $\mathcal{P} = \mathcal{P}^{-\circ}$，则称 $\mathcal{P}$ 是**正则**的。由等式 $\mathcal{P}^- = \mathcal{P}'^{\circ\prime}$，可以推出正则开集的另一个等价定义：开集 $\mathcal{P}$ 是正则的当且仅当 $\mathcal{P} = \mathcal{P}^{\perp\perp}$。

正则性保证了在建模过程中可以忽略低维度的特征，如孤立的点或线。

定义 2.2. 对一个集合 $\mathcal{S} \subset R^D$，若存在有限个解析函数 $g_i : R^D \to R$ 表示的区域

$$\chi_i = \left\{\mathbf{x} \in R^D : g_i(\mathbf{x}) \geq 0\right\}, \tag{2-1}$$

使得 $\mathcal{S}$ 为这些区域经过有限次布尔运算得到的结果，则称 $\mathcal{S}$ 为**半解析**的，g_i 为 $\mathcal{S}$ 的生成函数。（一个函数是解析的当且仅当函数定义域内任一点 $\mathbf{x_0}$ 的泰勒级数在某个邻域内都收敛到这个函数）

有了这些概念以后，我们可以给出三维殷集和殷空间的定义。

定义 2.3. 三维殷集：三维空间中边界有界的正则半解析开集。所有三维殷集构成的集合被称为殷空间，记为 $\mathbb{Y}$。

为了后续三维殷集边界表示定理的证明，我们还需给出一些说明。

记号 2.4. 在本文中，将不对流形和曲面作区分，并将三维空间中的二维有向紧流形简称为**紧曲面**。

定理 2.5. 二维紧流形分类定理：每一个紧曲面同胚于球面、轮胎面 *(torus)* 或是轮胎面的连通和 *(connected sum)*。

定义 2.6. 紧曲面将 R^3 空间分割为两个连通区域，一个有界，一个无界，分别称作**有界补集**和**无界补集**。其**定向**有正向和负向两种，正向紧曲面是指曲面上点的

外法向量由有界补集指向无界补集的紧曲面，反之则为负向紧曲面。给定一个有向黏合紧曲面，它的**内部**定义为外法向量的反方向指向的补集，**外部**定义为外法向量指向的补集。

我们还需定义两张曲面相交的方式。

定义 2.7. 对于两张曲面，若它们相交于某条交线 l，对于 l 的一个充分小的 r-邻域 $\mathcal{N}_r(l) = \{p \in R^3 : \exists q \in l, dist(p,q) < r\}$，$r$ 充分小，设两张曲面在 $\mathcal{N}_r(l)$ 内的部分分别为 $\mathcal{M}_1$，$\mathcal{M}_2$。则 $\mathcal{N}_r(l)\backslash\mathcal{M}_1$ 由两个不交的连通正则开集组成。若 $\mathcal{M}_2\backslash l$ 包含于这两个连通正则开集（不完全包含于其中一个），则称这两张曲面在 l 处为**恰当交**，否则称为**不当交**。若两张曲面相交于某点 p 或某张曲面片 sp，则称它们在 p 点或 sp 处不当交。若两张曲面没有交点，则称为**不交**的。

2.2 三维殷集边界表示

定义 2.8. 对于一个以一点 $p \in R^3$ 为球心的开球 $\mathcal{N}(p) \subset R^3$，$p$ 的**广义半径**是一条连接 p 和 $\partial\mathcal{N}(p)$ 上一点的简单曲线。

定义 2.9. 用 $\mathbb{D}$ 来表示 R^2 当中的闭单位圆盘，用 $\overline{\mathcal{N}(p)}$ 来表示以 $p \in R^3$ 为球心的一个开球的闭包。那么以 p 为中心的**广义圆盘**D 是 $\mathbb{D}$ 在满足 $p = f(0)$ 的连续映射 $f : \mathbb{D} \to \overline{\mathcal{N}(p)}$ 下的同胚像。

定义 2.10. 令 D 为以 p 为中心的广义圆盘，r_1，r_2 为 D 中两条不同的广义半径。那么 r_1 和 r_2 将 D 切分成两个连通分量，它们的边界都包含 r_1r_2。我们称这两个连通分量的闭包为**广义扇形**，r_1 和 r_2 为这些扇形的广义半径。如果 r 是广义扇形 F 的一条半径，那么称 F 是 r 上的广义扇形。

对每个广义半径 r。其上的广义扇形数量显然是偶数。

引理 2.11. 令 $p \in \partial Y$ 为殷集 $Y \subset R^3$ 上的一个边界点。对于任何充分小的开球 $\mathcal{N}(p)$，我们有

(a) $\partial Y \cap \mathcal{N}(p)$ 是有限个广义圆盘的并；

(b) (a) 中任意两个广义圆盘的交只能为 p 或两两只相交于 p 的广义半径的并；

(c) $\mathcal{N}(p) - \partial Y$ 由两两不交的正则开集组成，对于其中两个以某个广义扇形为公共边界的正则开集，一个是 Y 的子集而另一个是 $Y^{\perp}$ 的子集

证明：因为 Y 是半解析的，由定义 $\partial Y \cap \mathcal{N}(p)$ 通过有限个解析函数 $g_i : R^3 \to R$ 表示。而 $Y \cap \mathcal{N}(p)$ 为正则开集意味着 $\partial Y \cap \mathcal{N}(p)$ 不包含 0 维孤立点或 1 维曲线段。由隐函数定理，每个 $g_i(x, y, z) = 0$ 定义了一个曲面，所以 $\partial Y \cap \mathcal{N}(p)$ 只包含有限个开圆盘的同胚像，也就是广义圆盘，所以 (a) 得证。

由 Y 是半解析的，相交的连通分量必为有限个，那么总可以把 $\mathcal{N}(p)$ 的半径取的充分小，使得 $\mathcal{N}(p)$ 内广义圆盘相交的部分都包含 p，由正则性，相交部分不可能有二维的曲面，所以只能为 0 维的点或一维的曲线段，(b) 得证。

由 (a)(b)，我们通过从 $\mathcal{N}(p)$ 去掉 ∂Y 也就是一些广义圆盘得到有限个正则开集。对于任两个以同一广义扇形为边界的正则开集，它们如果都属于 Y，则违背了 Y 是正则的。类似的，它们也不能都属于 $Y^{\perp}$，所以 (c) 成立。

定义 2.12. $p \in Y$ 的**好邻域**是指满足引理 *2.11* 的邻域 $\mathcal{N}(p)$。

对于一个只有一条半径 r 的广义圆盘 F，我们可以再从 F 选取一条与 r 只相交于 p 的广义半径 r'，那么 r 和 r' 将 F 分为两个广义扇形。当一个好邻域 $\mathcal{N}(p)$ 给定以后，我们将上述引理中 $\partial Y \cap \mathcal{N}(p)$ 的广义圆盘简称为圆盘，将 $\mathcal{N}(p)$ 中圆盘相交出的广义半径和用来划分广义扇形的广义半径简称为半径，将 $\partial Y \cap \mathcal{N}(p)$ 中的广义扇形简称为扇形。

对于 $p \in \partial Y$ 的一个好邻域，我们记 $\mathcal{N}(p)$ 中的半径、圆盘、扇形的集合分别为 $\mathcal{R}(p)$，$\mathcal{D}(p)$ 和 $\mathcal{S}(p)$。

定义 2.13. 一点 $p \in \partial Y$ 被称为**非奇异**的当 $|\mathcal{D}(p)| = 1$，也就是说，p 点的充分小邻域内圆盘个数为 *1*；否则称 p 为**奇异点**。边界点 p 是**孤立奇异**的当它是奇异的且 $\mathcal{R}(p) = \emptyset$；一个点是**非孤立奇异**的当它是奇异的且 $\mathcal{R}(p) \neq \emptyset$。

注意到，p 是否奇异并不依赖于好邻域的选取。

推论 2.14. 当 p 是非孤立奇异的，任意 $\partial Y \cap \mathcal{N}(p) - \cup_{r \in \mathcal{R}(p)} r$ 中的点都是非奇异的，任意 $\cup_{r \in \mathcal{R}(p)} r$ 上的点都是非孤立奇异的。

引理 2.15. 一个殷集 $Y \subset R^3$ 满足

(a) ∂Y 只包含有限个孤立奇异点；

(b)∂Y 中的非孤立奇异点构成了一个紧的 *1* 维 *CW*-复形

证明：假设 ∂Y 包含无限多个孤立奇异点。由于 ∂Y 是紧的，存在一点 $y \in \partial Y$ 使得 y 的任意邻域都包含无限多个孤立奇异点。所以没有一个邻域是好邻域，这和引理 2.11 矛盾，所以 (a) 成立。

令 l 为所有非孤立奇异点的集合的一个连通分支，令 $p \in l$，$\mathcal{N}(p)$ 为 p 的一个好邻域，$\mathcal{R}(p)$ 为相应的广义半径集合。那么由推论 2.14，p 的一个好邻域与 l 的交同胚于 $\cup_{r\in\mathcal{R}(p)} r$。从而 l 局部同胚于图中的星，l 同胚于一个图，所以能赋予其 1 维 CW 复形结构。

为了说明非孤立奇异点的集合是闭的，我们可以通过证明任何收敛非孤立奇异点序列的极限仍然是非孤立奇异点。假设极限点 p 不是非孤立奇异的，那么它只能是非奇异的或者孤立奇异的。对于这两种情况，都存在 q 的充分小的邻域使得它不包含序列中的点，这与序列收敛性矛盾，所以非孤立奇异点的的集合一定是闭的，从而是紧的。

定义 2.16. 令 S 为拓扑空间，l_1 和 l_2 为 S 的子集。给定映射 $f : l_1 \to l_2$，将 S 关于 f 的 **商**（记为 S_f）定义为：对任意 $a \in l_1$，将 a 和 $f(a)$ 黏成一点。

定义 2.17. 折叠圆盘定义为将一个广义圆盘 D 沿着映射 $f : r_1 \cup \cdots \cup r_k \to l_1 \cup \cdots \cup l_t$ 黏合，其中，$r_1, \cdots, r_k$ 和 $l_1, \cdots, l_t$ 为 D 中的广义半径而 f 将每个 r_i 同胚地映射到某个 l_j。

定义 2.18. 广义扇形的**好配对**定义为将这些扇形两两配对，使得不存在两对 (F, F') 和 (G, G') 有恰当交。

引理 2.19. 令 $\mathcal{N}(p)$ 为一点 $p \in \partial Y$ 的好邻域。那么 $\partial Y \cap \mathcal{N}(p)$ 可以表示为有限个圆盘和折叠圆盘的并，并且它们蕴含了一个好配对。

证明：令 $r \in \mathcal{R}(P)$。由好邻域的定义，r 上存在偶数个扇形。注意到总有一个好配对存在：两个扇形构成一个配对仅当它们相邻。现在假设对于每个 $r \in \mathcal{R}(p)$，我们都有 r 上扇形的好配对。那么这些扇形自然地被分成若干组，每组构成一个扇形或折叠扇形：给定任意一个扇形 F_0，好配对给出了一个唯一的扇形序列 $F_0, F_1, \cdots, F_k, F_{k+1} = F_0$，其中 $F_i \neq F_j$ 除非 $\{i, j\} = \{0, k+1\}$ 并且 F_i 和 F_j 为 $r = F_i \cap F_j$ 上好配对给定的配对。如果从 F_0 到 F_k，存在一条半径经过了两次，那么 $\cup_{0\le i\le k} F_i$ 是一个折叠圆盘，否则为圆盘。由我们的构造，这些圆盘和折叠圆盘只在半径处相交，没有恰当交，所以是一个好配对。

定义 2.20. $\partial Y \cap \mathcal{N}(p)$ 的一个**好圆盘分解**为将 $\partial Y \cap \mathcal{N}(p)$ 按引理 *2.19* 的方式分解为圆盘和折叠圆盘。

引理 2.21. 令 p 和 q 为 ∂Y 上的非孤立奇异点，令 $\mathcal{N}(p)$ 和 $\mathcal{N}(q)$ 为 p 和 q 的好邻域。假设 $\mathcal{N}(p)$ 中的一条半径 r_1 与 $\mathcal{N}(q)$ 中的半径 r_2 相交。那么 $\mathcal{N}(p)$ 在 r_1 上的扇形与 $\mathcal{N}(q)$ 在 r_2 上的扇形存在一一对应，其中对应的扇形相交于二维子集。

证明：令 $x \in \mathcal{N}(p)$ 为 $r_1 \cap r_2$ 上一点，考虑 $\mathcal{N}(x)$ 上的一个好圆盘分解。通过把 $\mathcal{N}(x)$ 取得充分小，我们可以假设 $\mathcal{N}(x)$ 同时包含在 $\mathcal{N}(p)$ 和 $\mathcal{N}(q)$ 当中。注意到只要我们令 $\mathcal{N}(x)$ 充分小，$\mathcal{N}(x)$ 只有两条半径，都是 $r_1 \cap r_2$ 的子集。$\mathcal{N}(x)$ 中的每个扇形都包含在 $\mathcal{N}(p)$ 在 r_1 上的某个扇形当中，$\mathcal{N}(p)$ 在 r_1 上的每个扇形恰好包含一个 $\mathcal{N}(x)$ 的扇形。对于 $\mathcal{N}(q)$ 也有类似的结果。所以，存在 $\mathcal{N}(p)$ 中扇形和 $\mathcal{N}(q)$ 中扇形自然的一一对应，也即包含 $\mathcal{N}(x)$ 中同一扇形的两个扇形配对，它们的交显然是二维子集。

引理 2.22. 令 l 为 ∂Y 的非孤立奇异点集合的一个连通分支,令 $\{\mathcal{N}(y_1),\cdots,\mathcal{N}(y_k)\}$ 为覆盖 l 的有限个好邻域集。那么每个 $\mathcal{N}(y_i)$ 的好圆盘分解可按如下选取：对 l 中任意的 y 和 z，如果半径 $r_1 \in \mathcal{N}(y_i)$ 与 $r_2 \in \mathcal{N}(y_j)$ 相交，那么 r_1 上的扇形 F_1 和 F_2 构成一个好配对当且近当 r_2 上对应的扇形 F_1' 和 F_2' 构成好配对。

证明：对这些好邻域的每条半径，选择任意一个好配对。如果 $\mathcal{N}(y_1)$ 中的半径 r_1 与 $\mathcal{N}(y_1),\cdots,\mathcal{N}(y_k)$ 这些好邻域中的一个相交于 r_2，那么通过两个扇形集合间的一一对应，r_1 上扇形的好配对诱导了 r_2 上扇形的好配对。那么我们就可以用诱导出的好配对替换 r_2 已有的任意选择的好配对。对这些好邻域中所有半径重复这一个步骤，我们得到了所有扇形在所有半径的好配对。现在由引理 2.19，这些扇形的好配对诱导了每个好邻域当中的好圆盘分解，引理得证。

定理 2.23. ∂Y 同胚于将一系列二维紧曲面沿着与一维 *CW*-复形同胚的子集进行黏合得到的结果。

证明：我们先定义两个在有好圆盘分解的好邻域上的操作，即解开操作和分离操作。令 $y \in \partial Y$，$\mathcal{N}(y)$ 是一个好邻域，它有着好圆盘分解 $\partial Y \cap \mathcal{N}(y) = \cup_i D_i$，其中 D_i 是圆盘或者折叠圆盘。

对于 $\mathcal{N}(y)$ 中的每条半径，其上有 $2n$ 个扇形，n 是一个正整数，这些扇形有一个好配对。解开操作定义如下：将 r 分成 n 条半径 $r_1, \cdots, r_n$，这些半径只交于 y 点（可以看作在 r 上附加 n 个微小且不同的扰动），将 $2n$ 个扇形依次分配到这些半径上去，每条半径上分配一对构成好配对的扇形的并。所以解开操作的结果是 n 个“更大”的扇形。注意到这个操作是可以作用在 r 的任意小的邻域的，并且它并不会引入新交点。我们需要对每条半径依次进行解开操作，从而将折叠圆盘解开成了圆盘，得到的结果为若干个只相交于 y 点的圆盘。解开操作的几何意义是将好邻域内每条半径上的好配对扇形拉开一点，使得它们不再交于这条半径。

接下来我们使用分离操作。分离操作定义如下：对于解开操作的输出，也即一系列只相交于 y 点的圆盘，将这些圆盘在 y 点分离，使得这些圆盘两两不相交。分离操作与解开操作类似，可以通过在 y 上加微小且不同的扰动实现。通过解开和分离操作，好邻域内不再有奇异点。

令 l 为 ∂Y 中奇异点构成的集合的一个连通分支。令 $\mathcal{C}$ 为 l 上的有限点集，使得 $\{\mathcal{N}(y)|y \in \mathcal{C}\}$ 覆盖 l，其中 $\mathcal{N}(y)$ 是 y 的一个好邻域。通过选择合适的邻域，我们总可以使得每一个 $y \in \mathcal{C}$ 只属于这些邻域中的一个。选择满足引理 2.19 和引理 2.22 的好圆盘分解，则相交半径处来自不同好邻域的好配对是相容的。我们接下来进行解开操作和分离操作，好圆盘分解的选取和好邻域的选取都保证了我们的解开操作和分离操作在每一个好邻域上都是相容的。通过进行这些相容的操作，我们可以使得 l 上的点都不再是奇异点。

我们对每个 ∂Y 的连通分支都进行如上操作，得到的结果空间记为 $\widetilde{\partial} Y$。那么对于每个点 $y \in \widetilde{\partial} Y$ 都有一个邻域同胚于 R^2 空间中的开圆盘。因为 ∂Y 是紧的，所以 $\widetilde{\partial} Y$ 也是紧的。所以 $\widetilde{\partial} Y$ 是一个无边界的二维紧曲面（流形）。注意到 $\widetilde{\partial} Y$ 可能不是连通的，所以它同胚于若干个有有限亏格的闭合紧曲面的并。

解开操作和分离操作得到的结果都可以通过黏合操作来进行复原。解开操作得到的结果通过沿着半径重新黏合可以复原，分离操作得到的结果通过沿着点重新黏合可以复原。我们对上述 $\widetilde{\partial} Y$ 进行复原以后，则有定理的结果。

通过上述定理，我们已经证明了殷集的边界实际上是一些紧曲面沿着与 1 维 CW-复形同胚的子集进行黏合得到的结果。为了得到殷集边界的唯一表示，我们需要定义表示的基元，也就是黏合紧曲面。

定义 2.24. 黏合紧曲面是一个紧曲面或是这个紧曲面的商空间，其商映射将紧曲面沿与 *1* 维 *CW*-复形同胚的集合进行黏合，如果将这个集合删去，剩余的曲面片仍然是连通的。

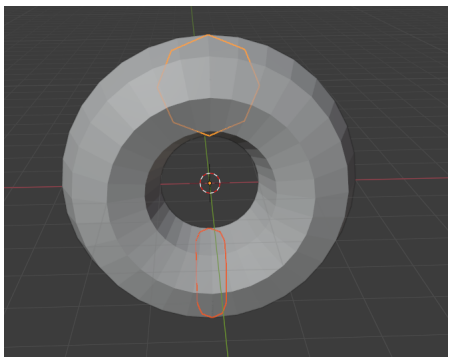


图 2.1: 主视图

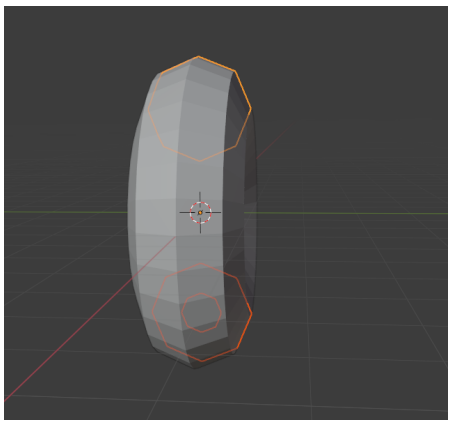


图 2.2: 左视图

图 2.3: 如图为一个大轮胎挖掉一个小轮胎和一个小球，小轮胎和小球分别与大轮胎交于一个圆，按照我们的定义，图中有三张黏合紧曲面：大轮胎为正向黏合紧曲面，小球和小轮胎为负向黏合紧曲面

记号 2.25. 由于紧曲面沿着黏合的与 *1* 维 *CW*-复形同胚的集合本质上就是黏合得到的曲面上的自交线，所以我们后面将这个集合称为自交线集，将集合中每个连通分量称为**自交线**。

定理 2.26. 黏合紧曲面在 R^3 中的补集为两个连通开集，一个有界，一个无界。

证明：由于紧曲面在 R^3 中的补集为一个有界连通开集和一个无界连通开集，只需证明将其黏合成黏合紧曲面的过程中，不会产生新的连通开集。若不然，紧

曲面沿着某个与 1 维 CW-复形同胚的子集黏合的过程中，产生了新的连通开集，那么将这个子集去掉，剩余的曲面片必然是不连通的，这就与黏合紧曲面的定义矛盾。所以黏合紧曲面在 R^3 中的补集也为一个有界连通开集和一个无界连通开集。

定义 2.27. 由定理 *2.24*，每个有向黏合紧曲面都与黏合前的有向紧曲面是一一对应的，黏合紧曲面的**定向**定义为与之对应的紧曲面的定向。正向黏合紧曲面上一点的外法向量由内部指向外部；负向黏合紧曲面则相反。给定一个有向黏合紧曲面，其**内部**定义为外法向量的反方向指向的补集，**外部**定义为外法向量指向的补集。

定义 2.28. 黏合紧曲面 G_1 **包含**G_2 当且仅当 G_1 的有界补集包含 G_2 的有界补集。

定理 2.29. 一个有界连通殷集的边界可唯一地分解为一个正定向黏合紧曲面和若干（可为 *0*）负定向黏合紧曲面，正定向黏合紧曲面包含所有负定向黏合紧曲面，负定向黏合紧曲面之间互不包含，任意两个黏合紧曲面之间无恰当交，交集与 *1* 维 *CW*-复形同胚。

证明：对于殷集的边界，我们逐条判断自交线，如果删去这条交线，剩余的曲面片不连通，我们就将它拆分成两张曲面，这里的拆分就是应用前面定义的解开和分离操作，对于拆分后的结果，递归地重复上述判断和操作，直到不能再拆分为止。由定义，我们拆分最终的结果就是一系列黏合紧曲面。每个黏合紧曲面根据其定向包含有界集或无界集，但都是连通的，无法继续拆分。由前面的讨论，我们可以将边界拆分成若干黏合紧曲面。我们先证明这些黏合紧曲面当中正定向的个数必定为 1。如果为 0，所有黏合紧曲面都是负定向的，那么将与有界性矛盾；如果大于等于 2，我们取出其中的两张正定向的黏合紧曲面，它们相交的集合与 1 维 CW-复形同胚，那么这两张黏合紧曲面的内部必然不连通，与连通性矛盾，所以个数只能为 1。由定理 2.23，黏合紧曲面之间是没有恰当交的。由殷集有界，所有负向黏合紧曲面必定包含在正向黏合紧曲面当中。对于任意两个负向黏合紧曲面，若存在包含关系，则被包含的黏合紧曲面不是边界，矛盾，所以负向黏合紧曲面之间互不包含。正向黏合紧曲面是该殷集的外表面，而负定向黏合紧曲面刻画了内部的洞，对于给定有界连通殷集，这些拓扑量都是唯一的，所以有界连通殷集边界的分解也唯一，得证。

对于无界连通殷集，结论也是类似的。

推论 2.30. 一个无界连通殷集的边界可唯一地分解为若干（可为 *0*）负定向黏合紧曲面，负定向黏合紧曲面之间互不包含，且无恰当交，任意两个黏合紧曲面之间的交集与 *1* 维 *CW*-复形同胚。

通过对连通殷集的边界分解，我们可以直接得到它的闭包的 Betti 数 B_0 和 B_2 。

推论 2.31. 一个连通殷集的连通分量个数为 *1*，洞的个数为负定向黏合紧曲面的个数。

为了便于讨论，我们给出殷集边界表示的记号。

记号 2.32. 对于一个有界连通殷集 Y，由定理 *2.29* 导出的边界表示记为 $\mathcal{G}^+ = \left\{G_1^+, G_2^-, \cdots, G_n^-\right\}$，其中 G_1^+ 为正向黏合紧曲面，G_i^- 为负向黏合紧曲面。对于一个无界连通殷集 Y，由推论 *2.30* 导出的边界表示记为 $\mathcal{G}^- = \left\{G_1^-, G_2^-, \cdots, G_n^-\right\}$，其中 G_i^- 为负向黏合紧曲面。根据这个记号，我们有 $Y = \cap_{G_i \in \mathcal{G}} int(G_i)$，其中 *int* 根据黏合紧曲面的定向来取内部。

现在，我们可以建立一般殷集的边界表示理论。

定理 2.33. 一个有界殷集的边界可唯一地分解为 $\mathcal{G}_Y = \cup_{i=1}^{n} \mathcal{G}_i^+$，连通分量个数为 *n*，洞的个数为所有 $\mathcal{G}_i^+$ 中负向黏合紧曲面个数的和；一个无界殷集的边界可唯一地分解为 $\mathcal{G}_Y = \cup_{i=1}^{n} \mathcal{G}_i^+ \cup \mathcal{G}_0^-$，连通分量个数为 *n+1*，洞的个数为所有 $\mathcal{G}_i^+$ 和 $\mathcal{G}_0^-$ 中负向黏合紧曲面个数的和，其中，$\mathcal{G}_i^+$ 和 $\mathcal{G}_0^-$ 分别为有界连通殷集和无界连通殷集的边界表示。

证明：对于殷集的每个连通分量，应用定理 2.29，推论 2.30 和 2.31，即可得到唯一的边界表示和 Betti 数，再利用记号 2.32，可得到定理中的表示。

定义 2.34. 定义黏合紧曲面空间 $\mathbb{G} := \{\mathcal{G}_Y : Y \in \mathbb{Y}\}$，特别的，我们用 $\hat{0}$ 和 $\hat{1}$ 分别表示 $\mathbb{Y}$ 中空集和全集对应的黏合紧曲面集。

定义 2.35. 定义取内部算子 $\rho : \mathbb{G} \to \mathbb{Y}$ 如下：

$$\rho(\mathcal{G}_Y) = \cup_{\mathcal{G}_i \subset \mathcal{G}_Y}^{\perp\perp} \cap_{G_j \in \mathcal{G}_i} int(G_j). \tag{2-2}$$

特别地，有 $\rho(\hat{0}) = \emptyset$ 和 $\rho(\hat{1}) = R^3$. 由定理 *2.33*，$\mathbb{G}$ 和 $\mathbb{Y}$ 是一一对应的。

3 三维殷集布尔代数算法

3.1 理论部分

在这一节将会介绍我们的三维殷集布尔代数算法，在此之前，先给出两个重要的定理，这是建立布尔代数的基础。

定理 3.1. 设 $\mathbb{B}$ 表示一个拓扑空间 $\mathcal{X}$ 中所有正则开集构成的集合，并定义 $\mathcal{P} \cup^{\perp\perp} \mathcal{Q} := (\mathcal{P} \cup \mathcal{Q})^{\perp\perp}$，则 $B_R := (\mathbb{B}, \cup^{\perp\perp}, \cap, \perp, \emptyset, \mathcal{X})$ 是一个布尔代数。

证明：见文献[10] 的第十章。

定理 3.2. 代数 $B_Y := (\mathbb{Y}, \cup^{\perp\perp}, \cap, \perp, \emptyset, R^3)$ 是一个布尔代数。

证明：由定义 2.3，殷集是正则开集，那么根据定理 3.1，它满足所有布尔运算的运算律，为了证明殷空间关于这些布尔运算构成布尔代数，我们只需证明它关于这些运算封闭。根据定义 2.2，半解析集在布尔运算作用下是封闭的，而边界有界的集合经过布尔运算也仍然边界有界，所以殷空间关于布尔运算封闭，定理得证。

定理 3.2 从理论上保证了三维殷集布尔代数的可行性，然而，直接对三维殷集内部的点进行布尔运算，内存和时间的消耗都是巨大的。为了提高布尔运算的效率，我们利用定理 2.33 和定义 2.34，在黏合紧曲面空间 $\mathbb{G}$ 上建立布尔运算，并且我们将利用取内部算子 ρ 建立黏合紧曲面空间 $\mathbb{G}$ 与殷空间 $\mathbb{Y}$ 的同构，从而将三维殷集上的布尔代数转移到它的边界空间上的布尔代数，提高效率。

首先，我们需要定义三个运算，分别是切割运算、黏合运算和拆分运算，这三个运算是布尔运算设计的核心。

算法 3.3. 切割运算：

输入：黏合紧曲面集 SG，交线集 IS

输出：切割完的有向曲面片集 SP

算法步骤：

(1) 对 $G \in SG$，取出 IS 中所有属于 G 的交线，将 G 沿这些交线进行切割，得到若干有向曲面片，将这些有向曲面片添加到 SP 当中。

算法 3.4. 黏合运算：

输入：有向曲面片集 SP

前置条件：SP 中的有向曲面片恰好是由某个殷集 $\mathcal{Y}$ 的边界 $\partial\mathcal{Y}$ 沿某交线集切割得到的

输出：闭合曲面集 SS

算法步骤：

(1) 遍历 SP，对于已经闭合的有向曲面片，将其初始化为黏合紧曲面，添加到 SS 当中；

(2) 从 SP 中取出一张有向曲面片 s_{in}，对 s_{in} 的每一条边界曲线 b，取其上一点 p 的充分小邻域，曲面在这一小邻域内近似平面，可求得法向量 n_b（称作关于 p 的法向量），b 在 s_{in} 中定向由 n_b 做右手螺旋决定。从 SP 当中寻找边界包含 b 且 b 在其中定向与在 s_{in} 中定向相反的有向曲面片，对于满足条件的有向曲面片 s_c，其关于 p 的法向量记为 n_c，计算 n_b 和 n_c 的夹角 β（$0 < \beta < \pi$），则 s_{in} 和 s_c 在 b 处的夹角 $angle = \pi - \beta$。如果 $n_b \times n_c$ 与 b 在 b_0 点的方向向量点乘小于 *0*，令 $angle = 2\pi - angle$，我们计算所有满足条件的有向曲面片的 *angle*，选择其中最小的，对应的有向曲面片记为 s_{out}，将 s_{in} 和 s_{out} 沿 b 进行黏合，得到 γ。重复以上操作直到 γ 闭合，将闭合曲面 γ 添加到 SS 当中，删掉 SP 中对应的曲面片；

(3) 重复 *(2)* 直到 SP 空。

事实上，算法 3.4 的 (2) 中得到的每张闭合曲面都是某个连通殷集的边界，且闭合曲面之间无恰当交。

算法 3.5. 拆分运算：

输入：闭合曲面集 SS

输出：黏合紧曲面集 SG

算法步骤：

(1) 从 SS 中取出一张闭合曲面 S，求出 S 的所有自交线，添加到自交线集 IS 中；

(2) 将 S 沿 IS 进行切割，得到有向曲面片集 SP；

(3) 若 SP 的元素个数为 *1*，将 S 初始化为黏合紧曲面并添加到 SG 中，算法终止；

(4) 将 SP 中所有曲面片定向反向；

(5) 调用算法 *3.4*，将 SP 黏合得到黏合紧曲面集 SG；

(6) 将 SG 中所有黏合紧曲面定向反向；

(7) 重复 *(1)-(6)* 直到 SS 空。

定理 3.6. 给定有向曲面片集 SP，满足其中的有向曲面片恰好是由某个殷集 $\mathcal{Y}$ 的边界 $\partial\mathcal{Y}$ 沿某交线集切割得到的，那么对 SP 调用算法 *3.4* 和算法 *3.5*，将会得到一系列黏合紧曲面，这些黏合紧曲面恰好构成了 $\mathcal{Y}$ 的唯一边界表示。

证明：我们首先证明每张闭合曲面经过算法 3.5 都会分解为若干无恰当交的黏合紧曲面。给定闭合曲面 S，如果 S 已经是黏合紧曲面，显然成立；若 S 不是黏合紧曲面，由算法 3.4 的 (2)，它是某个连通殷集 $\mathcal{Y}_1$ 的边界。若 $\mathcal{Y}_1$ 是有界的，由定理 2.29，S 是由一张正向黏合紧曲面和若干负向黏合紧曲面黏合得到的。将 $\mathcal{Y}_1$ 求补集，$\mathcal{Y}_1$ 的每个有界补集都变为不含洞的有界连通殷集，而不含洞的有界连通殷集的边界可唯一地分解为一张正向黏合紧曲面。所以在算法 3.5 中，负向黏合紧曲面 G_i^- 切割得到的所有曲面片在翻转定向后黏合得到的曲面必然是正向黏合紧曲面 G_i^+（与 G_i^- 只差一个定向），在 (5) 中对 G_i^+ 的定向求反即得到 G_i^-，所以每张负向黏合紧曲面都能正确黏合得到。剩余的曲面片只能黏合为 S 分解得到的正向黏合紧曲面。由算法 3.4 的 (2)，这些黏合紧曲面之间无恰当交，所以得证。对于 $\mathcal{Y}_1$ 无界的情形，证明也是类似的。

其次，对每张黏合曲面都调用算法 3.5，最终得到了一系列两两之间无恰当交的黏合紧曲面，它们构成了某个殷集的边界表示。由于这些黏合紧曲面的并等于 $\partial\mathcal{Y}$，且黏合紧曲面上每一点的外法向量方向都与 $\partial\mathcal{Y}$ 上对应点的外法向量方向相同，所以它们构成了 $\mathcal{Y}$ 的唯一边界表示。

定理 3.6 表明了算法 3.4 加算法 3.5 是算法 3.3 的逆运算。有了以上三个运算，我们现在可以定义三维殷集的求补运算和求交运算。

算法 3.7. 求补运算 $'$：

输入：殷集 Y 的边界对应的黏合紧曲面集 $\mathcal{G}_Y$

输出：$Y^\perp$ 的边界对应的黏合紧曲面集 $\mathcal{G}_{Y^\perp}$

算法步骤：

(1) 将 $\mathcal{G}_Y$ 中的黏合紧曲面两两求交，得到交线集 IS；

(2) 调用算法 *3.3*，对 $\mathcal{G}_Y$ 按 IS 进行切割运算，得到有向曲面片集 SP；

(3) 将 SP 中的所有有向曲面片方向取反；

(4) 调用算法 *3.4*，对 SP 进行黏合运算得到闭合曲面集 SS；

(5) 调用算法 *3.5*，对 SS 进行拆分，得到 $\mathcal{G}_{Y^\perp}$ 并输出。

引理 3.8. 算法 *3.7* 定义的补运算 $'$ 满足：

$$\forall \mathcal{G} \in \mathbb{G}, \quad \rho(\mathcal{G}') = (\rho(\mathcal{G}))^\perp \tag{3-1}$$

证明：对 $\forall \mathcal{G} \in \mathbb{G}$，由定理 3.6，$'$ 是封闭的，所以 $\mathcal{G}' \in \mathbb{G}$。只需证明，$\rho(\mathcal{G}') = (\rho(\mathcal{G}))^\perp$。殷集 $(\rho(\mathcal{G}))$ 和 $(\rho(\mathcal{G}))^\perp$ 有着相同的边界，而 $\mathcal{G}$ 与 $\mathcal{G}'$ 中的点也是相同的，所以 $(\rho(\mathcal{G}))^\perp$ 的边界点和 $\mathcal{G}'$ 中的点是相同的。我们只需要保证 $\mathcal{G}'$ 的所有黏合紧曲面上点的外法向量方向与 $\mathcal{G}$ 中对应点的外法向量方向是相反的即可，而这个是由算法 3.7 的第 (3) 步保证的，所以结论成立。

在介绍求交运算之前，我们先给出两个引理，这是求交运算算法设计的基础。

引理 3.9. 记 $Y := int(G_1) \cap int(G_2)$，其中 G_1 和 G_2 为有向黏合紧曲面，对于曲面 β 满足 $\beta \subseteq G_1$ 且 $\beta \subset R^3 \backslash G_2$，我们有 $\beta \subset \partial Y$ 当且仅当 $\beta \subset int(G_2)$。对于曲面 β 满足 $\beta \subseteq G_1 \cap G_2$，我们有 $\beta \subset \partial Y$ 当且仅当 β 在 G_1 中的定向与在 G_2 中的定向相同。

证明：我们只证明前半部分，因为后半部分的证明是类似的。由定理 3.2，Y 是一个殷集。要证明 $\beta \subseteq \partial Y$，我们只需证明对 $\forall p \in \beta$，对 p 的充分小邻域 $\mathcal{N}(p)$，同时包含 Y 中的点和 $Y^\perp$ 中的点。由于 $p \in \beta$ 且 $\beta \subseteq G_1$，所以 $\mathcal{N}(p) \cap \text{int}(G_1) \neq \emptyset$，又因为 $\beta \subset \text{int}(G_2)$，所以 $\mathcal{N}(p) \cap \text{int}(G_2) \neq \emptyset$，所以 $\mathcal{N}(p) \cap Y \neq \emptyset$。又由 $\beta \subset G_1$，$\mathcal{N}(p) \cap \text{int}(G_1)^\perp \neq \emptyset$，所以 $\mathcal{N}(p) \cap Y^\perp \neq \emptyset$。综上，对 $\forall p \in \beta$，$\mathcal{N}(p)$ 同时包含 Y 中的点和 $Y^\perp$ 中的点，引理得证。

由引理 3.9 可以直接推得：

引理 3.10. 记 $Y := \rho(\mathcal{G}_1) \cap \rho(\mathcal{G}_2)$，其中 $\mathcal{G}_1$ 和 $\mathcal{G}_2$ 为黏合紧曲面空间 $\mathbb{G}$ 的元素，如果曲面 β 是 $\mathcal{G}_1$ 中某张黏合紧曲面 G_i 的子集且 $\beta \subset R^3 \backslash \partial(\rho(\mathcal{G}_2))$，就有 $\beta \subset \partial Y$ 当且仅当 $\beta \subset \rho(\mathcal{G}_2)$。如果曲面 β 既是 $\mathcal{G}_1$ 中某张黏合紧曲面 G_i 的子集，也是 $\mathcal{G}_2$ 中某张黏合紧曲面 G'_j 的子集，我们有 $\beta \subset \partial Y$ 当且仅当 β 在 $\mathcal{G}_1$ 中的定向与在 $\mathcal{G}_2$ 中的定向相同。

算法 3.11. 求交运算 $\wedge$：

输入：殷集 Y_1 和 Y_2 的边界对应的黏合紧曲面集 $\mathcal{G}_{Y_1}$ 和 $\mathcal{G}_{Y_2}$

输出：$Y_1 \cap Y_2$ 的边界对应的黏合紧曲面集 $\mathcal{G}_{Y_1 \cap Y_2}$

算法步骤：

(1) 将 $\mathcal{G}_{Y_1}$ 中的黏合紧曲面与 $\mathcal{G}_{Y_2}$ 中的黏合紧曲面两两求交（不对 $\mathcal{G}_{Y_1}$ 或 $\mathcal{G}_{Y_2}$ 内的两张黏合紧曲面求交），得到交线集 IS*(*包含重合曲面的信息*)*；

(2) 调用算法 *3.3*，对 $\mathcal{G}_{Y_1}$ 与 $\mathcal{G}_{Y_2}$ 按 IS 进行切割运算，得到有向曲面片集 SP_1 与 SP_2；

(3) 将 SP_1 与 SP_2 中重合的有向曲面片取出，如果方向相同，则添加其中任意一张到有向曲面片集 SP，无论方向是否相同都从 SP_1 与 SP_2 中删去这张有向曲面片；

(4) 对 SP_1 中的有向曲面片 s，如果 s 上除边界以外的点构成的集合包含于 Y_2，则将 s 添加到 SP；

(5) 对 SP_2 中的有向曲面片 s，如果 s 上除边界以外的点构成的集合包含于 Y_1，则将 s 添加到 SP；

(6) 调用算法 *3.4*，将 SP 进行黏合运算得到闭合曲面集 SS；

(7) 调用算法 *3.5*，对 SS 进行拆分，得到 $\mathcal{G}_{Y_1 \cap Y_2}$ 并输出。

引理 3.12. 算法 *3.11* 定义的交运算 $\wedge$ 满足：

$$\forall \mathcal{G}, \mathcal{H} \in \mathbb{G}, \quad \rho(\mathcal{G} \wedge \mathcal{H}) = (\rho(\mathcal{G})) \cap \rho(\mathcal{H}). \tag{3-2}$$

证明：对 $\forall \mathcal{G}, \mathcal{H} \in \mathbb{G}$，由引理 3.10，算法 3.11 中 (3)(5)(6) 对 SP 的选取保证了其中曲面片的并恰好等于 $(\rho(\mathcal{G})) \cap \rho(\mathcal{H})$ 的边界，所以由定理 3.6，$\mathcal{G} \wedge \mathcal{H} \in \mathbb{G}$。由于算法 3.11 没有改变曲面片的定向，$\mathcal{G} \wedge \mathcal{H}$ 中的点的外法向量和 $\rho(\mathcal{G})) \cap \rho(\mathcal{H})$ 的边界点的外法向量方向相同，所以 $\rho(\mathcal{G} \wedge \mathcal{H}) = (\rho(\mathcal{G})) \cap \rho(\mathcal{H})$。

定义 3.13. $\mathbb{G}$ 上的求并运算 $\vee : \mathbb{G} \times \mathbb{G} \to \mathbb{G}$ 定义如下：

$$\forall \mathcal{G}, \mathcal{H} \in \mathbb{G}, \quad \rho(\mathcal{G} \vee \mathcal{H}) := (\mathcal{G}' \cap \mathcal{H}')'. \tag{3-3}$$

求并运算可以根据 DeMorgan 律，由求交和求补运算直接得到。

定理 3.14. 布尔代数 $(\mathbb{G}, \vee, \wedge, ', \hat{0}, \hat{1})$ 与 $(\mathbb{Y}, \cup^{\perp\perp}, \cap, \perp, \emptyset, R^3)$ 在取内部算子 ρ 的映射下是同构的。

证明：由定义 2.35 和 3.13，引理 3.8 和 3.12 以及 DeMorgan 律可得。

定理 3.14 保证了我们可以直接对殷集的边界进行布尔运算，下面我们将给出具体的算法细节。

3.2 具体实现

我们在实际操作时，输入为殷集边界分解得到的一些黏合紧曲面，这些黏合紧曲面往往是三角剖分的形式，黏合紧曲面的定向决定了三角形边的遍历顺序是顺时针还是逆时针（三角形边的遍历顺序按右手螺旋规则产生的外法向量与黏合紧曲面的外法向量一致）。无论是求交还是求补算法，我们首先都要对黏合紧曲面进行两两求交。两张黏合紧曲面的求交，实际是对这两张黏合紧曲面中的所有三角形两两求交，每一个三角形都记录对应的交线和重合的信息。

给定两个三角形 triA 和 triB，求交分为两种情形：(1) 当 triA 和 triB 共面的时候，我们先将它们投影到合适的坐标平面，得到 triA′ 和 triB′，将 triA′ 中的每条边 e 都与 triB′ 进行求交（可以转化为 e 所在直线分别与 triB′ 的三条边求交），将 triB′ 中的每条边也都与 triA′ 进行求交，将结果还原回原空间并记录下来，如果存在交点，这两个三角形的关系记为重合；(2) 当 triA 和 triB 不共面的时候，我们首先求出当 triA 所在平面和 triB 所在平面的交线 l，将 triA 与 l 投影到合适的坐标平面进行求交，再将 triB 与 l 投影到合适的坐标平面进行求交 (两个坐标平面未必相同)，将两次求交的结果还原回原空间并合并。

在所有三角形求交完毕后，为了后续的切割和黏合，我们需要将每一个三角形根据它记录的交线重新进行三角剖分。我们的三角剖分算法如下：给定三角形，我们将它的所有交线的端点和三角形的顶点都添加到顶点集当中，每次从顶点集取两个顶点，如果它们的连线与已有的所有交线、添加的连线、三角形的边都不重合且没有除端点以外的交点，就将这两个顶点连起来，重复以上操作直到所有顶点对判断完毕。最终得到的即为三角形关于给定交线的一个三角剖分，对那些剖分后与交线重合的三角形的边，我们额外标记为**交线边**。

接下来需要将黏合紧曲面沿交线进行切割，也即算法 3.3，这在具体实现时更像是一种黏合。

算法 3.15. 切割算法实现：

输入：黏合紧曲面集 $\mathcal{G}$ 的三角剖分包含的所有三角形 *vecTri*

前置条件：*vecTri* 已经关于交线重新进行三角剖分

输出：$\mathcal{G}$ 中的黏合紧曲面沿交线切割得到的曲面片集 SP

算法步骤：

(1) 从 *vecTri* 中任取一个三角形 *tri*，遍历 *tri* 的三条边，如果不是交线边，就继续搜索与这条边相邻的三角形；如果是交线边则换下一条边，我们将搜索到的所有三角形从 *vecTri* 中删去，初始化为一个曲面片 p，并添加到 SP 当中；

(2) 重复 *(1)* 直到 *vecTri* 为空，输出 SP。

我们的切割算法实际上是一个深度优先搜索，从一个小三角形出发将搜索到的相邻三角形黏合起来，当遇到了交线边，则交线的相邻三角形就不黏合，最终实现的效果就是将黏合紧曲面沿交线切割成曲面片。

黏合运算和拆分运算的实现与算法 3.4 和 3.5 类似，这里不多赘述。下面将讨论最后一个难题，判断曲面片是否在殷集内部，这个问题的本质难题在于判断一个点与一张黏合紧曲面的位置关系，算法如下：

算法 3.16. 判断一个点与一张黏合紧曲面的位置关系：

输入：R^3 中一点 p，一张黏合紧曲面对应的三角形集 *vecTri*

前置条件：*vecTri* 中三角形边的遍历顺序决定了三角形的外法向量 n（对三角形任意两条边 e_1 和 e_2，若顺序为从 e_1 到 e_2，则 $e_1 \times e_2$ 的方向与 n 的方向相同），n 与黏合紧曲面在这一点的外法向量方向相同

输出：*1*（在黏合紧曲面有界补集内），*0*（在黏合紧曲面上），*-1*（在黏合紧曲面无界补集内）

算法步骤：

(1) 从 p 出发，任作一条射线 l，将 l 与 *vecTri* 中所有三角形求交。如果没有交点，返回*-1*；

(2) 选取与 p 距离最短的交点 q，如果 p 和 q 重合返回 *0*；

(3) 设 q 对应的三角形为 *tri*，如果 q 在 *tri* 内部，设 *tri* 的外法向量为 n，计算 $dot(n, q-p)$，如果大于 *0* 则返回 *1*，小于 *0* 则返回*-1*；

(4) 重复 *(1)(2)(3)*。

这个算法成立的原因在于从 p 出发的射线与黏合紧曲面的所有交点当中，我

们选取的是最近的交点 q，线段 pq 上除 q 以外的点同属于一个连通分量，所以它们与黏合紧曲面的位置关系是一样的，如果向量 $q-p$ 与 q 所在三角形外法向量的夹角小于九十度，则 $q-p$ 必然属于黏合紧曲面内部；如果大于九十度，则 $q-p$ 属于黏合紧曲面外部；不存在等于九十度的情况，因为 q 是三角形的内点，并且是射线上离 p 最近的点。

我们的算法在设计过程中都遵循一个重要的规则：

定义 3.17. 不确定性参数的唯一准则：当两个点的距离小于用户给定的容忍度 ϵ 的时候，我们将其视为同一个点。

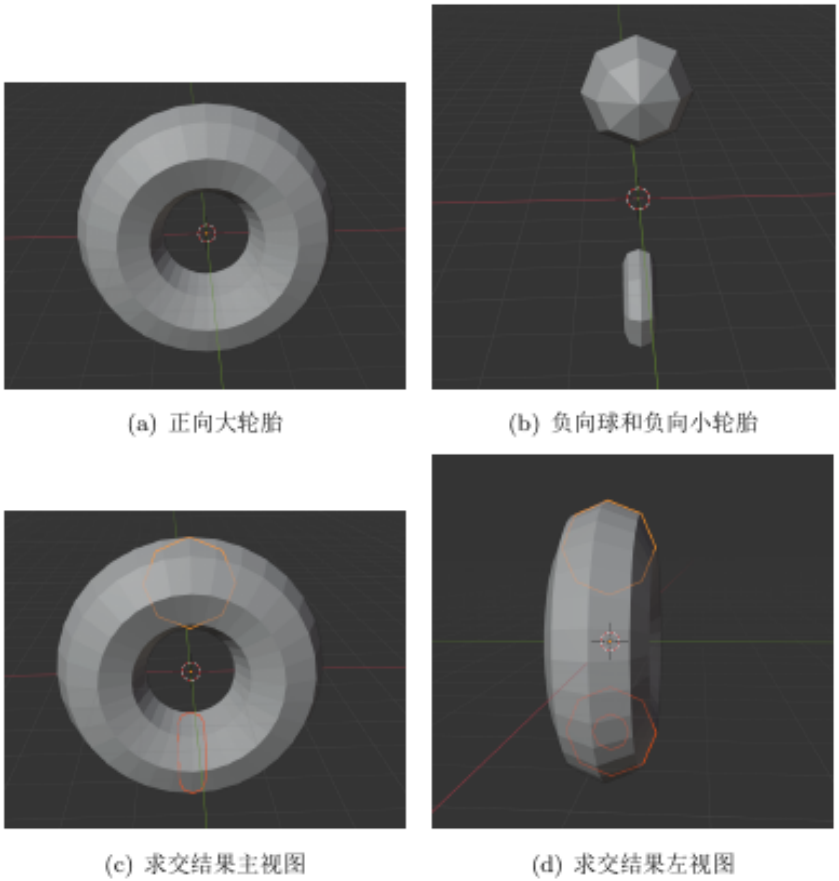
不确定性参数的唯一准则的意义在于：(1) 浮点系统是有误差的，对于同一个点，在计算过程中坐标可能会有差异，这一准则可以让我们忽略这些差异，保证算法的正确运行；(2) 线段求交点是计算几何当中很困难的一个问题，当两条线段的斜率充分接近时，计算出的交点的误差会非常巨大。而这一准则可以使得我们在线段求交时，将这种病态的情形视为平行的情况来求交点，对于斜率相差较大的线段才使用通常的求交，从而使得我们的计算误差减小，程序更稳定。

我们下面将讨论如何将这一准则应用到程序当中。首先，我们定义了一个顶点比较函数 PointCompare(Point a，Point b)，这个函数会判断点 a 和点 b 的距离是否小于 ϵ。在所有需要添加顶点或是判断点与点之间关系的算法，我们都会调用这个函数。比如，在三角剖分时，我们会构造一个顶点集，这在 C++ 当中是一个 set，它的比较函数是我们定义的 PointCompare 函数，这保证了顶点集中的点之间的距离都是大于 ϵ 的。利用 PointCompare 函数，我们可以定义 SegmentCompare 函数，用来比较两条线段是否相等，它会判断两条线段的两个端点是否相等，如果两条线段的两个端点分别相等（距离小于 ϵ），则我们认为这两条线段也重合。在判断平行关系的时候，比如线段是否平行，对于线段 s_1 和 s_2，设其对应的向量为 $\vec{s_1}$ 和 $\vec{s_2}$，我们计算 $norm(\vec{s_1}\times\vec{s_2})/norm(\vec{s_2})$，其中 norm() 函数返回给定向量的模长，当结果小于 ϵ 时，我们认为 s_1 和 s_2 平行。对于程序中涉及数字比较的算法，我们也会引入这一准则，比如在算法 3.16 的第 (3) 步当中，实际的程序是如果大于 ϵ 返回 1，小于 ϵ 返回-1，这可以保证程序运行时忽略浮点误差带来的影响。

4 测试结果

本节将会展示我们的三维殷集布尔代数程序在一些测试集上的交并结果。

首先是一个简单的例子，我们有两个殷集，其中一个殷集只有一张正向的大轮胎面，另一个殷集由一张负向的小球面和一张负向的小轮胎面构成。大轮胎面与小球和小轮胎都分别相交于一个一维圆。我们对这两个殷集求交，得到的结果如图 4.1，等价于一个大轮胎挖去了小球和小轮胎，输出的黏合紧曲面有 3 张，分别为正向的大轮胎面、负向的小球面、负向的小轮胎面，这与我们的定理 2.29 和定理 2.33 是吻合的。



(a) 正向大轮胎　(b) 负向球和负向小轮胎

(c) 求交结果主视图　(d) 求交结果左视图

图 4.1: 测试一

我们现在将小球和小轮胎用一根细管子连起来，变成一张负的黏合紧曲面，再与正的大轮胎进行求交，如图 4.2，结果仍是正确的，我们的求交程序输出的

结果为一张正的黏合紧曲面和一张负的黏合紧曲面。

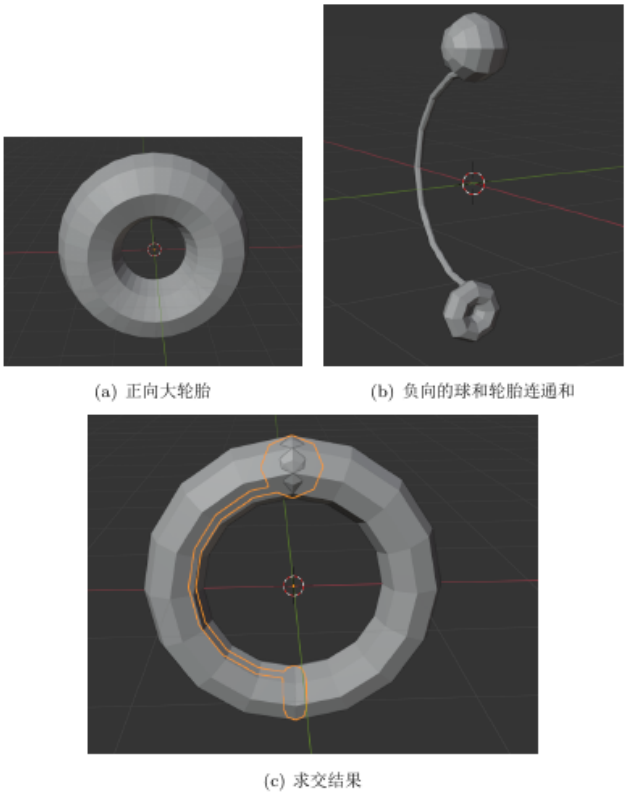
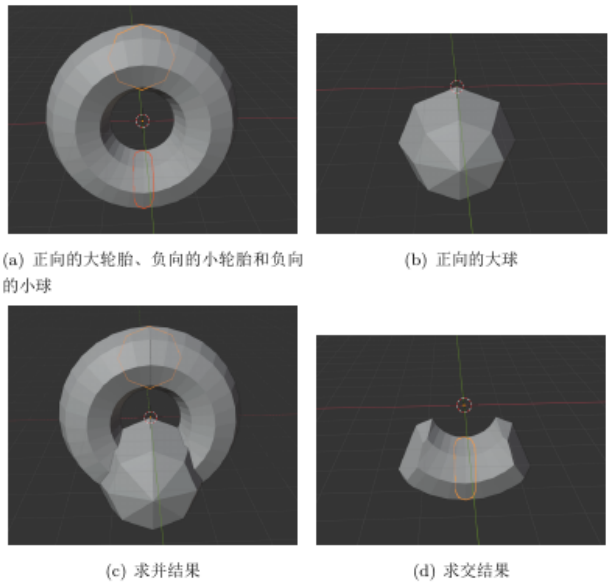


图 4.2: 测试二

再考虑一个更复杂的情形，我们现在将图 4.1 中的求交结果作为输入，再输入一个正向的大球，对这两个殷集进行求交和求并运算，结果呈现在图 4.3 当中。我们的求并结果输出两张黏合紧曲面，分别是正向的大轮胎和大球求并完重新黏合后的结果与负向的小球。我们的求交结果输出两张黏合紧曲面，分别是正向的大轮胎和大球求交完重新黏合后的结果与负向的小轮胎，都与我们的理论相一致。

现在我们将大球移到上方，结果如图 4.4，求并结果仍是类似的，但是求交结果有了本质的区别，这是因为求交以后，有两个连通分量，而我们的算法正确的处理了这种情况，输出了两张正向黏合紧曲面。注意到，这两张黏合紧曲面中间都挖掉了半个球，这是轮胎面片和球面片正确黏合的结果。

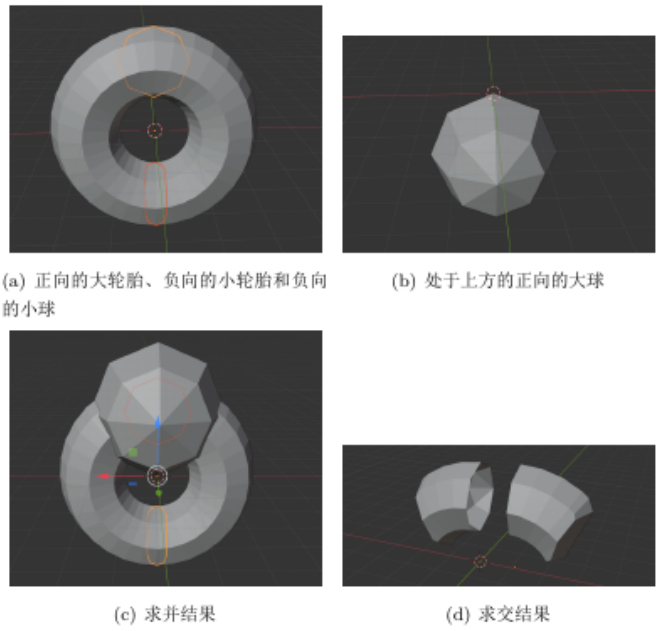
最后给出两个有趣的例子。首先，我们将一张正向的兔子曲面与一张正向的



(a) 正向的大轮胎、负向的小轮胎和负向的小球　(b) 正向的大球

(c) 求并结果　(d) 求交结果

图 4.3: 测试三



(a) 正向的大轮胎、负向的小轮胎和负向的小球　(b) 处于上方的正向的大球

(c) 求并结果　(d) 求交结果

图 4.4: 测试四

球面进行求并运算，见图 4.5，得到了一张正向黏合紧曲面，实现了兔子坐在球上的效果。

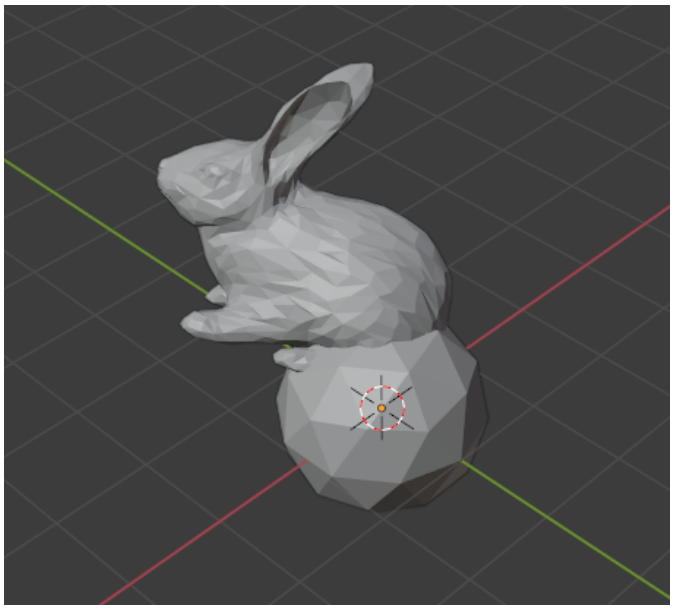


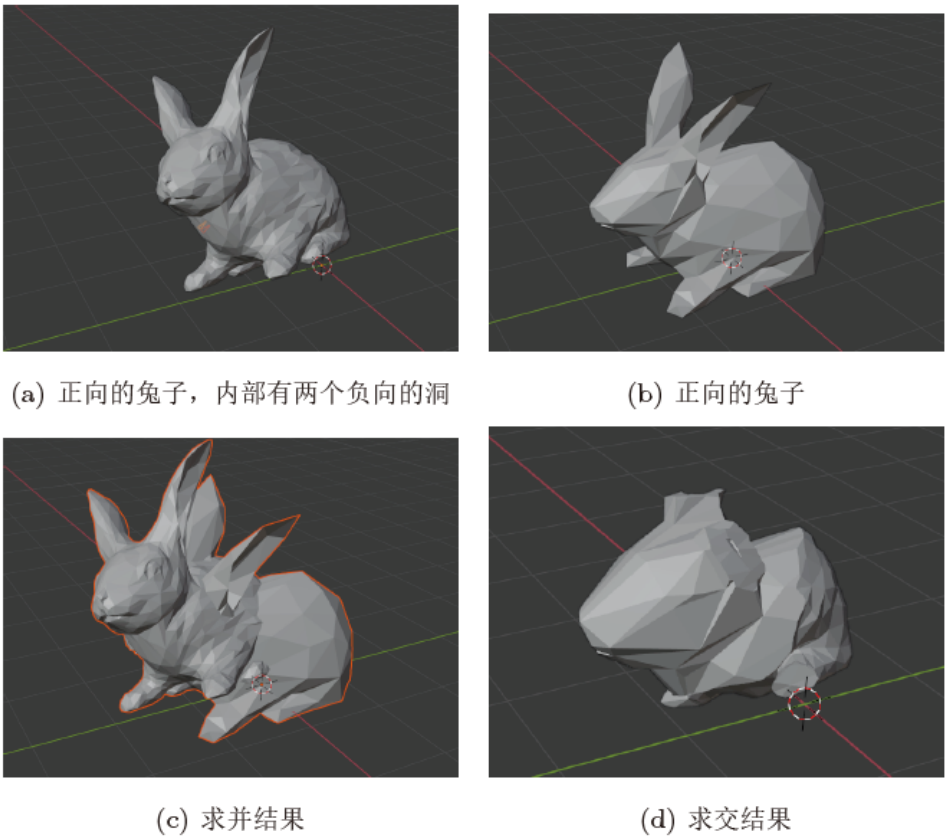
图 4.5: 测试五

另一个例子是将两只兔子求交和求并。其中，第一只兔子由三张黏合紧曲面构成：一张正向的兔子曲面，内部有两张负向的黏合紧曲面，表示的殷集同胚于三维球挖掉两个洞；第二只兔子只有一张正向的兔子曲面，表示的殷集同胚于三维球。求并以后只有一张正向的黏合紧曲面，求交以后为一张正向的黏合紧曲面和内部的两张负向黏合紧曲面，结果见图 4.6。

5 结论与展望

为了给三维多相流的研究提供整套的分析和建模工具，我们在二维殷集的基础上进行推广，完成了三维殷集的边界表示理论，将三维殷集的边界唯一分解为若干黏合紧曲面，由这些黏合紧曲面可以直接得到殷集连通分量个数和洞的个数。我们以边界表示理论为基础提出高效的布尔代数算法，我们用 C++ 实现了提出的算法，并在多个数据集都得到了正确的交并结果。我们希望能将这一工作不仅仅是应用到多相流领域，也能应用到其它强调拓扑和几何结构的领域，我们希望写的软件是用户友好的，从而可以为科研工作者们的研究提供便利。

未来的工作需要解决的问题有：(1) 目前只能求出殷集闭包的 Betti 数 B_0 和 B_2，我们希望未来能够高效地求解 B_1；(2) 不确定性参数唯一准则在算法的各



(a) 正向的兔子，内部有两个负向的洞　(b) 正向的兔子

(c) 求并结果　(d) 求交结果

图 4.6: 测试六

个部分还没有做到相容，当遇到一些极端情况时程序仍会出错；(3) 目前程序的输入只能是殷集，我们希望能写一个函数，将不是殷集的数据集自动改造成殷集，从而提高程序的适用范围；(4) 目前采用的求交算法还是简单的两两求交，时间复杂度是 $O(N^2)$，未来我们将会采用更高效的扫描面算法，时间复杂度仅为 $O(NlogN)$，同时，也会采用更高效的三角剖分算法。

6 参考文献

[1] REQUICHA A, VOELCKER H. Boolean operations in solid modeling: Boundary evaluation and merging algorithms[J]. Proceedings of the IEEE, 1985, 73(11): 30-44.

[2] ROSSIGNAC J R, REQUICHA A A G. Solid modeling[J]. Georgia Institute of Technology, 1999.

[3] STEUER S. Methods for polygonalization of a constructive solid geometry description in web-based rendering environments[J]., 2012.

[4] HOFFMANN C M. Geometric and solid modeling[M]. [S.l.]: Morgan Kaufmann Publishers Inc., 1989.

[5] REQUICHA A. Representations of rigid solid objects[M]. [S.l.]: Springer, 1980.

[6] KACZYNSKI T, MISCHAIKOW K, MROZEK M. Computational homology[M]. [S.l.]: Springer, 2004.

[7] COQUILLART S, STRAßER W, STUCKI P. From object modelling to advanced visual communication[M]. [S.l.]: Springer, 1994.

[8] ZHANG Q, LI Z. Boolean algebra of two-dimensional continua with arbitrarily complex topology[J]. Math. Comp. 89 (2020), 2020: 2333-2364.

[9] DELFINADO C, EDELSBRUNNER H. An incremental algorithm for Betti numbers of simplicial complexes on the 3-sphere[J]. Computer Aided Geometric Design, 1995, 12.

[10] GIVANT S. Introduction to Boolean Algebras[M]. [S.l.]: Springer, 2009.

本科生毕业论文（设计）任务书

一、题目：

二、指导教师对毕业论文（设计）的进度安排及任务要求：

起讫日期 20　年　月　日　至　20　年　月　日

指导教师（签名）_____ 职称 _____

三、系或研究所审核意见：

负责人（签名）_____

年　月　日

本科生毕业论文（设计）考核

一、指导教师对毕业论文（设计）的评语：

指导教师（签名）_____

年　月　日

二、答辩小组对毕业论文（设计）的答辩评语及总评成绩：

成绩比例	文献综述（10%）	开题报告（15%）	外文翻译（5%）	毕业论文质量及答辩（70%）	总评成绩
分值	**10**	**15**	**5**	**70**	**100**

负责人（签名）_____

年　月　日

第二部分

毕业论文开题报告

浙 江 大 學

本 科 生 毕 业 论 文

文献综述和开题报告

姓名与学号	邱云昊 3170102133
指导教师	张庆海
年级与专业	2017级数学与应用数学
所在学院	数学科学学院

一、题目：三维殷集的布尔代数算法和程序实现

二、指导教师对文献综述、开题报告、外文翻译的具体要求：

指导教师（签名）_____

年　月　日

一、文献综述

1 背景介绍

1.1 实体建模

有物理意义的区域（在均匀连续介质意义下）是无处不在的，对这种区域的建模在很多工程和科学应用中有着基础而重要的意义。通常，对这种有物理意义区域的建模是一个被称为实体建模的成熟领域的研究目标。三维实体建模是一组将三维有物理意义区域用于数学和计算机建模的一致的准则，它包含了聚焦“信息完全”的实体表示的一整套理论、技术和系统[1]，并允许所表示实体的几何性质能够自动被计算。它与几何建模、计算机图形学等相关领域的主要差别在于它强调物理的保真性。实体建模是一项非常吸引人的技术，为一些本需要人来手工操作的任务，如制图、有限元分析的网格生成等，提供了自动化的工具。实体建模是三维计算机辅助设计（CAD）的基础，它支持物理对象的数字模型的创造、交换、可视化、动画制作、查询和注释。它在三维计算机图形学、计算机视觉、机器人学等学科领域同样发挥着巨大的作用。三维实体建模的运用提高了复杂零件设计的速度和灵活性，在许多行业，如医疗、娱乐等，都有着广泛的应用[2]。

2 国内外研究现状

2.1 研究方向及进展

2.1.1 构造实体几何

构造实体几何（CSG）方法[3] 将实体定义为：基元进行一系列运算（如求交、求并）得到的结果。这里的基元是一些预先给定的参数化的实体，如正方体、球、圆柱等等。基元和它们经过运算得到的结果可以进行平移和伸缩。一个 CSG 实体的描述往往是精确的，可以对它进行参数化和编辑，使得它适用于程序化、高

层次的建模。但是它并不携带任何关于相关实体连通性甚至存在性的信息。事实上，一个非空的 CSG 描述可以表示一个空的实体，比如对两个不相交的实体求交。为了解决这个问题，一种思路是把边界考虑进来，这引出了实体建模的另一个主流方法，边界表示法（BREP）[3]。

2.1.2 边界表示法

边界表示法（BREP）的核心思想是：一个实体的边界可以通过面的集合来表示，每一个面可以通过边的集合来表示，每一条边可以通过顶点来表示。边界表示法强调实体外表的细节，它详细记录了实体的所有几何信息和拓扑信息，将面、边、顶点分层记录，建立层与层之间的联系。为了保证 BREP 表示的实体是合法的，它的边界必须是嵌入在三维空间的可定向闭合流形[4]。BREP 中表示边界的面往往是一些多边形，被称为多边形网格。三角形是最简单的多边形，所有多边形都可以分割成若干三角形，因此，它是 BREP 中最常用的。由三角形面组成的 BREP 被称为三角网格，也就是代数拓扑中的三角剖分，它在研究中被广泛地使用，尤其是在 GPU 当中，这种数据结构上的运算非常高效。

2.1.3 其他实体建模表示方法

CSG 和 BREP 是实体建模最常用的两种方法，对于其它的方法，基于[2] 和[5]，我们在这里做一个简略的介绍。参数化基元实例化方法预先给定一个包含多族基元的库，每一族的基元通过确定若干参数后变成实例，也即实体的表示。例如，库中的螺栓基元，通过确定它的长度、宽度等参数可以表示所有的螺栓。这种方法并不能将实例进行组合产生新的更复杂的实例。空间枚举法将建模空间划分成固定大小的网格，通常是立方体网格，每一个网格称为单元，会用一个点比如网格的中心来表示。实体由它所占据的空间内的所有网格构成的集合表示。单元分解法是一般化的空间枚举法，区别在于每一个单元的大小不再是固定的，它可以看作是以立方体为基元的 CSG 的一个特例。单元分解法要求不相邻的实体只能进行求并操作，从而避免了 1.1.1 中提到的非空 CSG 描述表示一个空的实体的情况。扫描法是一种适合运动规划的方法，它预先给定一个运动集合和轨迹，实体由运动集合沿轨迹扫过的体积表示。这种表示在计算两个实体相对运动

时的动态干涉、机器人导航等应用背景下特别有用。

2.1.4 同调代数

同调代数是一种非常强大的工具，它可以通过对局部数据的计算得到空间的全局特征，它将代数、组合、计算和拓扑很好地结合在一起[6]。同调代数方法可以与 BREP 和 CSG 结合，例如，在 CSG 中，当基元为立方体时，它所表示的实体就是一个立方复形，就可以依此构建链复形，计算同调群，从而它的连通分量、洞的个数都可以得到。BREP 中对边界的三角剖分可以看作是构造了一个单纯复形，也可以作同样的计算得到 Betti 数。

2.1.5 布尔代数和 r-sets

建模空间上需要定义一些操作，这是建模过程中不可或缺的一部分。毕竟，建模的主要目的是为了回答关于建模对象的一些问题，而这通常涉及对建模对象的操作。从理论而言，建模空间需要与问题相适应，从计算而言，这些操作需要给出代数化、构造化的定义，从而可以将其设计成具体的算法。

布尔运算是实体建模过程中重要而有用的操作，大部分实体建模方法都会支持布尔运算。布尔运算在实体建模中的作用包括[1]：可以对一些已有的简单实体进行有限次布尔运算得到具有复杂拓扑的新的实体；能够检测空间中物体的碰撞和干扰；能够建模和模拟制造过程，比如铣削和钻孔。借助布尔运算，可以刻画拓扑变化，自然地处理其中的退化情形。布尔运算设计的一个基本的要求是对象在布尔运算的作用下是封闭的，这对于通常集合论中的交并补是不一定成立的，必须将其正则化，与此相关的是 r-sets 的理论。r-sets[7] 是欧式空间中有界的正则半解析闭集。它的正则性捕捉到了连续介质中低维特征的缺失；半解析则保证了边界的行为良好。r-sets 精确的数学定义保证了可以在其上建立理论正确的布尔运算算法，使得 r-sets 在布尔运算下是封闭的[2]，从而为实体建模中的一些重要的表示方法如构造实体几何等奠定基础。

2.1.6 二维殷集

用二维殷集进行建模是最近提出的一种实体建模方法[8]。二维殷集是 R^2 中边界有界的正则半解析的开集。上述定义中，正则性保证了在建模时可以忽略那些低维度的信息，很好地刻画了有物理意义的区域；半解析和边界有界性保证了殷集在计算机中是能够表示的，是可计算的，这些定义与 r-sets 是类似的。二维殷集与 r-sets 的区别在于，殷集要求集合是开的，开集保证了殷集边界表示的唯一性，从而能够建立殷空间上布尔代数和边界表示空间上布尔代数之间的同构。所以，殷集是连续介质的恰当数学模型。

在[8] 中，证明了二维殷集边界的拓扑结构定理，从而每一个殷集的边界可唯一地分解为若干两两几乎不交且分段解析的一维有向约当曲线。事实上，殷集可以看作是 r-sets 和 BREP 的结合与推广。殷集上定义了高效而简单的布尔运算，可以捕捉拓扑变化、处理退化情形，并且由它的边界表示可以非常自然地在常数时间内得到实体的连通分量和洞的个数。殷集是由一些可计算的数学性质定义的，独立于特定的表现形式或应用程序，可以很好地应用于多相流领域和其它需要利用几何和拓扑信息进行严格分析的领域。

2.2 存在问题

现有的 CSG、BREP 等三维实体建模的表示方法，或是忽略了模型的拓扑信息和几何信息，或是需要较多的内存才能计算得到拓扑信息，效率不高，这并不适用于多相流的研究。在计算连通分量个数、洞的个数等信息时，无论是基于立方复形还是单纯复形，时间复杂度都至少为 O(n)[6, 9]，n 为三角形或立方体的个数，因为至少需要将这些基元遍历一遍。当拓扑变化发生时，会在边界上产生非流形点，已有的布尔代数算法很难处理这些非流形点。r-sets 要求实体是闭的，这对于某些领域（如多相流）的研究是不自然的，并且会导致实体边界表示的不唯一。此外，r-sets 上建立的布尔运算算法非常复杂且效率不高。二维殷集将 r-sets 中要求的闭集条件改为开集后，则可以使边界有唯一的表示，在此基础上建立更简单高效的布尔代数算法。然而二维殷集并不能简单地推广到三维，三维的情况更加复杂，需要重新证明相关定理并设计布尔运算的算法。

3 研究展望

三维殷集作为二维殷集的推广，情况更加复杂，但也保留了二维殷集的好的性质：

（1）三维殷集是对三维有物理意义空间的建模，可以捕捉到实体建模相关的特征；

（2）三维殷集是数学化、可计算的定义，独立于特定的表现形式，所以它不仅可以应用于实体建模，也可以应用于多相流领域等强调几何和拓扑结构的领域；

（3）可以证明，三维殷集的边界能唯一地表示为一些有向黏合紧曲面，这与二维的情形是相似的，我们可以依此建立简单高效的布尔运算算法。三维殷集可以捕捉到拓扑的变化，能处理出现非流形点这种退化的情形；

（4）三维殷集是由二维有向黏合紧曲面来表示，本质上是一种 BREP，但是通过边界表示可以非常自然地得到 betti 数如连通分量的个数和洞的个数，时间复杂度仅为常数。

三维殷集为重视几何和拓扑建模的领域的研究奠定了非常坚实的理论基础，同时，提高了计算的效率。我们希望在三维殷集现有的理论基础上，完成以下工作：

（1）我们将会设计黏合映射和切割映射的算法，并在此基础上设计三维殷集的布尔运算算法；我们将会证明我们的算法可以建立三维殷空间到二维黏合紧曲面空间上的同构；

（2）我们将会用 C++ 实现三维殷集及其上布尔代数的软件包，使得用户能够方便地使用殷集进行建模；

（3）我们将会设计不确定性参数准则，将足够接近的两点看成同一点，并将这种原则一致地反映在我们的算法和程序当中。

4 参考文献

[1] REQUICHA A, VOELCKER H. Boolean operations in solid modeling: Boundary evaluation and merging algorithms[J]. Proceedings of the IEEE, 1985, 73(11): 30-44.
[2] ROSSIGNAC J R, REQUICHA A A G. Solid modeling[J]. Georgia Institute of Technology, 1999.
[3] STEUER S. Methods for polygonalization of a constructive solid geometry description in web-based rendering environments[J]., 2012.
[4] HOFFMANN C M. Geometric and solid modeling[M]. [S.l.]: Morgan Kaufmann Publishers Inc., 1989.
[5] REQUICHA A. Representations of rigid solid objects[M]. [S.l.]: Springer, 1980.
[6] KACZYNSKI T, MISCHAIKOW K, MROZEK M. Computational homology[M]. [S.l.]: Springer, 2004.
[7] COQUILLART S, STRAßER W, STUCKI P. From object modelling to advanced visual communication[M]. [S.l.]: Springer, 1994.
[8] ZHANG Q, LI Z. Boolean algebra of two-dimensional continua with arbitrarily complex topology[J]. Math. Comp. 89 (2020), 2020: 2333-2364.
[9] DELFINADO C, EDELSBRUNNER H. An incremental algorithm for Betti numbers of simplicial complexes on the 3-sphere[J]. Computer Aided Geometric Design, 1995, 12.

二、开题报告

1 问题提出的背景

1.1 三维空间有物理意义区域的建模

三维空间中有物理意义的区域（均匀连续体的意义下）是非常普遍的，这种区域的建模在工程和科学领域有着非常广泛的应用。传统中，对有物理意义区域的建模是实体建模这一成熟的研究领域的主要目标。然而，过去在多相流的研究中，由于计算资源和理论的局限性，通常不对流体区域进行建模。如今，由于多相流领域的快速发展，需要对流体区域进行建模，从而可以研究一些更复杂的现象，如涉及到流体中拓扑变化的问题。

1.2 布尔代数

建模过程中不可或缺的一部分是对建模空间赋予的操作，毕竟，建模的主要目的是回答关于被建模对象的问题，这些问题通常需要对建模对象进行某些操作。从理论上而言，建模空间需要与问题相适应，而从计算上而言，这些操作需要被代数且构造地定义，从而可以实现在合理时间内完成的算法。

布尔运算是实体建模中过程中重要且有用的操作，大部分可行的建模方法都会支持一些布尔运算。布尔运算在实体建模中的作用包括[1]：可以对已有的简单实体进行有限次交并补布尔运算得到具有复杂拓扑的新的实体；能够检测空间中物体的碰撞和干扰，刻画拓扑变化，自然地处理其中的退化情形；能够建模和模拟制造过程，比如铣削和钻孔。对于布尔运算的一个基本的要求是对象在布尔运算的作用下是封闭的，这对于通常集合论中的交并补是不一定成立的，必须将布尔运算正则化。

1.3 殷集

定义：殷集 $\mathcal{Y}$ 是有界正则半解析的开集。所有这样的殷集构成了殷空间 Y。

上述定义中，正则性保证了在建模时可以忽略那些低维度的信息，很好地刻画了有物理意义的区域；半解析和有界性保证了殷集在计算机中是能够表示的，是可计算的；开集保证了殷集边界表示的唯一性，使得能够建立殷空间和其边界空间之间的同构。所以，殷集是连续介质的恰当数学模型。

在[8]中，证明了二维殷集边界的拓扑结构定理，从而每一个殷集的边界可唯一地由若干几乎不交分段解析的一维有向约旦曲线表示，因此构造了一个从包含若干约旦曲线的 realizable spadjors 组成的约旦空间到二维殷集构成的殷空间之间的双射 ρ,ρ 将约旦曲线按定向映成其内部。并且在约旦空间上定义和实现了交并补算法，从而实现了二维殷集上的布尔代数到约旦空间上的布尔代数的同构，将二维的交并补简化到一维，使得算法更加高效。

对于三维殷集的情形，也已经有边界拓扑结构定理证明了其边界可以唯一地分解为有限个两两几乎不交的二维有向黏合紧曲面，同样的，可以找到从包含若干黏合紧曲面的 realizable spadjors 组成的黏合紧曲面空间到三维殷集构成的殷空间之间的双射 ρ，ρ 将黏合紧曲面按定向映成其内部。然而，还没有现成的算法可以实现黏合紧曲面空间上的交并补运算。

2 本研究的意义和目的

本次研究，希望在三维殷集的边界拓扑结构定理的基础上，设计关于黏合紧曲面的切割和黏合算法，进而设计黏合紧曲面空间上的交并补算法。我们将会用 C++ 设计一个软件包，实现黏合紧曲面空间上的交并补算法。我们将证明取内部算子 ρ 是关于黏合紧曲面空间上布尔运算到三维殷空间上布尔运算的同态，从而证明三维殷空间和黏合紧曲面空间是同构的。如此，三维的交并补就被简化到二维，极大地提高了效率。同时，我们设计的算法保证了我们能够精确地刻画拓扑的变化和处理退化的情形。

3 项目的主要内容和技术路线

3.1 主要研究内容

根据三维殷集的边界拓扑结构定理，我们讨论的空间变成了黏合紧曲面空间，其中的元素为 realizable spadgors。每个 realizable spadgor 由若干两两几乎不交的二维有向黏合紧曲面构成。在这个空间上，我们将设计切割映射，对于每个 realizable spadgor，按照其内黏合紧曲面之间的不当交线和非流形点，切割成一些带边有向曲面片。我们将设计黏合映射，将这些曲面片按照一定规则黏合成一个 realizable spadgor。黏合映射可看作切割映射的逆。

在黏合映射和切割映射的基础上，我们将设计黏合紧曲面空间上的布尔运算——交并补的算法。对于补运算，首先利用切割映射将 realizable spadgor 切割成若干带边有向曲面片，将这些曲面片的定向改变，再用黏合映射重新黏成一个 realizable spadgor。对于交运算，我们将两个 realizable spadgors 内的黏合紧曲面按照其不当的交线和非流形点进行切割，对于得到的有向带边曲面片，我们将设计规则对其进行筛选，得到一个曲面片集，最后再对曲面片集进行黏合得到一个 realizable spadgor，即为它们的交。并运算可直接由交运算和补运算得到。

我们将会证明我们所设计的交并补算法是合理的，即取内部映射 ρ 是关于交和并运算的同态，且 ρ 和补运算可交换。从而我们可以证明黏合紧曲面空间上的布尔代数与三维殷集空间上的布尔代数在 ρ 作用下是同构的。

3.2 技术路线

在用 C++ 编程实现时，得到的输入为一系列曲面片，这些曲面片为三角剖分的形式，其实是一些单纯复形，我们要求这些曲面片确实构成了一个殷集的边界。这样做的一个好处是后续的研究如样条插值时需要用到三角形网格，这里的三角剖分可以为之奠定基础。我们所设计的切割映射和黏合映射在具体实现时，输入都为单纯复形。

在实现布尔运算时，有两个非常基础并且重要的问题需要被解决：第一个是两个二维连通紧曲面对应的三角剖分的求交；第二个是判断一个顶点是否在二维有向紧曲面对应的三角剖分的有界补集内部。为了解决第一个问题，我们需要

解决两个线性三角形的求交，我们将会设计算法来解决这个问题。第二个问题对应的一维情形即判断一个点是否在简单多边形内部是计算几何中著名的问题，可以通过扫描线算法高效地解决，而在二维情况下，我们希望用与一维类似的办法去解决。

我们的算法区别于其他布尔算法的一个关键在于用户可以输入一个不确定性参数用来控制算法中的不确定度。更明确的，给定不确定性参数 $\epsilon > 0$，我们定义两点为同一点如果它们之间的距离小于 ϵ。我们需要将这个准则推广到线的重合与面的重合，并反映在我们所有的算法中，使得整个程序的所有模块都是一致的。比如在线性三角形求交时，当两个线性三角形所在平面之间的夹角足够接近时，我们认为它们是平行或共面的。这种准则的实行会使得我们的程序非常鲁棒，能够高效地处理各种刻画拓扑变化的退化情形。在计算几何中，两条线段或是两个平面的相交可能会不可避免地产生与自身不一致的结果并导致程序中止，这一困难的数学核心在于两条线段或是两个平面在没有约束的长度尺度下相交时产生的病态。我们的准则为它们的相交在尺度上给予了限制，从而避免了这一病态的发生。在实际应用中，往往存在一个尺度，我们并不需要长度小于这个尺度的信息，所以我们的不确定性准则对于一个合适选取的 ϵ，既保证了信息的完整性，同时也保证了程序的鲁棒性和效率。

3.3 可行性分析

我们所研究的三维殷集上的布尔代数在工程和科学领域有着很大的应用前景。比如，在多相流研究中，用殷集对流体进行建模，并利用布尔运算，可以捕捉到拓扑的变化；利用黏合紧曲面的定向，我们可以在常数时间内得到研究对象的 Betti 数。殷集作为建模工具，能够稳定且高效地获取研究对象的拓扑性质，是现有方法所缺少的，也是继续深入研究所必需的。

对于用户而言，并不需要明确黏合面之间的包含关系，可以将合紧曲面以任意顺序输入，只需要保证输入的确实是一个 realizable spadgor 即可，我们的程序在初始化时会通过排序算法将所有黏合紧曲面组织成一个 realizable spadgor。同时，用户可以自由选择输出的类型，既可以和输入类型一样，也可以根据包含关系输出一个 Hasse 图。我们设计的软件包简单高效且有着友好的用户接口。

4 研究计划进度安排及预期目标

4.1 进度安排

1 月 29 日 - 3 月 1 日　掌握理解已有的三维殷集的理论部分，学习现有的二维殷集布尔代数的程序。

3 月 1 日 - 4 月 15 日　设计黏合紧曲面空间上的切割映射、黏合映射算法和布尔运算算法，用 C++ 完成各种算法的实现，并封装成一个软件包

4 月 15 日 - 5 月 3 日　撰写毕业论文

4.2 预期目标

希望能够理解并建立一个完整且严密的三维殷集及其上布尔代数的理论；希望能用 C++ 基本实现设计的布尔运算算法，对于各种退化非退化的情形都能适用；希望能将不确定性准则运用到各个算法中，使之与各个算法相容；希望将各个算法封装在一个有着友好用户接口的软件包中，使得这个软件包能被相关专业的科研工作者使用。

5 参考文献

[1] REQUICHA A, VOELCKER H. Boolean operations in solid modeling: Boundary evaluation and merging algorithms[J]. Proceedings of the IEEE, 1985, 73(11): 30-44.
[2] ROSSIGNAC J R, REQUICHA A A G. Solid modeling[J]. Georgia Institute of Technology, 1999.
[3] STEUER S. Methods for polygonalization of a constructive solid geometry description in web-based rendering environments[J]., 2012.
[4] HOFFMANN C M. Geometric and solid modeling[M]. [S.l.]: Morgan Kaufmann Publishers Inc., 1989.
[5] REQUICHA A. Representations of rigid solid objects[M]. [S.l.]: Springer, 1980.
[6] KACZYNSKI T, MISCHAIKOW K, MROZEK M. Computational homology[M]. [S.l.]: Springer, 2004.
[7] COQUILLART S, STRAßER W, STUCKI P. From object modelling to advanced visual communication[M]. [S.l.]: Springer, 1994.
[8] ZHANG Q, LI Z. Boolean algebra of two-dimensional continua with arbitrarily complex topology[J]. Math. Comp. 89 (2020), 2020: 2333-2364.
[9] DELFINADO C, EDELSBRUNNER H. An incremental algorithm for Betti numbers of simplicial complexes on the 3-sphere[J]. Computer Aided Geometric Design, 1995, 12.

三、外文翻译：具有任意复杂拓扑的二维连续体上的布尔代数

摘要

为了为二位连续体的复杂拓扑、大型几何变换、拓扑变化如多相流背景下的合并的研究奠定坚实的理论基础，我们提出了一个数学模型。我们的建模空间殷集，由有限边界的正则半解析开集构成，并且被赋予构造的和代数的布尔代数的定义。我们模型主要显著的特征包括：（a）流体的拓扑信息如 Betti 数可以很简单地在常数时间内得到，（b）流体的拓扑变化可以被边界上的非流形点捕捉，（c）流体的布尔运算可以正确地处理各种退化情形并适用于任意复杂拓扑，但是它们又是简单而高效的因为它们只涉及判断一个点和一条约旦曲线的相对位置以及几条曲线段的相交。最后，通过将殷空间和最近的 cubic MARS 方法结合用于追踪单个涡流中的复杂流体，（a）和（c）的实用性得以证明。

1 简介

均匀连续体意义下的有物理意义区域是普遍存在的，它们的建模在无数的科学与工程的应用中具有重要的意义。传统上，对有物理意义区域的建模是实体建模这一成熟研究领域的主要研究课题。相比之下，在多相流领域，流体的建模一直是被避免的。然而，随着多相流学科的飞速发展，一直在呼吁建立这样的模型使得复杂的现象，比如流体的拓扑变化，可以被严格地研究。

在这篇文章中，我们主要解决这一需求，通过在多相流中引入流体建模，类似于计算机辅助设计中的实体建模。我们提出了一个拓扑空间用于流体建模，并为它配备了自然的代数结构，从而可以获取重要的拓扑信息以及实现简单高效的布尔运算。

这这项工作中，我们致力于建立流体建模的理论基础。为了避免重新造轮子，我们试图利用实体建模的已有结果，但是发现并不适用于流体建模。基于胞腔复形的拓扑计算需要很多内存，效率难以被接受。r-set 也不适合流体建模：首

先需要 r-set 为闭集并不适合节目追踪方法的数值分析；其次，拓扑变化导致的边界上的非流形点使得边界表示不唯一，从而使得殷空间到约旦空间的映射不是同构的；第三，r-set 可以被修改，产生一个实现非常简单高效的布尔代数。

2 殷集

正则半解析集：一个开集是正则的当且仅当它等于它的闭包的内部，一个闭集是正则的当且仅当它等于它的内部的闭包；一个集合 $\mathcal{S} \subseteq \mathbb{R}^{\mathrm{D}}$ 是半解析的当存在有限个解析函数 $g_i : \mathbb{R}^{\mathrm{D}} \to \mathbb{R}$ 使得 $\mathcal{S}$ 是以下集合

$$\mathcal{X}_i = \left\{ \mathrm{x} \in \mathbb{R}^{\mathrm{D}} : g_i(\mathrm{x}) \geq 0 \right\}$$

经过有限次布尔运算得到的。

殷集：殷集 $\mathcal{Y} \subseteq \mathbb{R}^2$ 是边界有界的正则半解析开集。

殷空间：所有殷集构成的类为殷空间 $\mathcal{Y}$。

定理 2.1：代数 $\mathrm{Y}:=\left(\mathbb{Y}, \cup^{\perp\perp}, \cap, \perp, \emptyset, \mathbb{R}^2\right)$ 是一个布尔代数。

约旦曲线及定向：约旦曲线是平面上没有自交点的闭合曲线。一条约旦曲线是正定向的当且仅当沿着它的方向行进时，它所包围的有界部分始终在它的左边；否则，该约旦曲线为负定向。

定理 2.2：对于一个连通的殷集 $\mathcal{Y} \neq \emptyset, \mathbb{R}^2$，它的边界可以唯一地分解为有限个两两几乎不交的约旦曲线。

定理 2.3：假设一个殷集 $\mathcal{Y} \neq \emptyset, \mathbb{R}^2$ 是连通的。那么它的边界的唯一约旦曲线分解可以被唯一定向，使得

$$\mathcal{Y} = \bigcap_{\gamma_j \in \mathcal{J}_{\theta y}} int\left(\gamma_j\right).$$

$\mathcal{J}_{\partial Y}$ 作为构成 $\mathcal{Y}$ 边界的定向约旦曲线集合，只可能是以下两种类型

$$\begin{cases} \mathcal{J}^- = \left\{\gamma_1^-, \gamma_2^-, \ldots, \gamma_{n_-}^-\right\}, & n_- \geq 1 \\ \mathcal{J}^+ = \left\{\gamma^+, \gamma_1^-, \gamma_2^-, \ldots, \gamma_{n_-}^-\right\}, & n_- \geq 0 \end{cases}$$

其中所有 γ_j^- 是负定向的，互相不能被包含。对于 $\mathcal{J}^+$，我们也有

$$\forall j = 1, 2, \ldots, n_-, \quad \gamma_j^- \prec \gamma^+$$

推论 2.4：每个殷集 $\mathcal{Y} \neq \emptyset, \mathbb{R}^2$ 可以唯一地表示为

$$\mathcal{Y} = \bigcup_{j}^{\perp\perp} \bigcap_{i} int\left(\gamma_{j,i}\right)$$

其中 j 是 $\mathcal{Y}$ 的连通分支序号而 $\gamma_{j,i}$ 是两两几乎不交的定向约旦曲线。一个连通殷集中洞的个数等于它的上述唯一表示中负定向约旦曲线的个数；一个有界殷集中连通分支的个数等于它的上述唯一表示中正定向约旦曲线的个数。

3 殷集上的布尔代数

定义 3.1：以下边界到内部算子 ρ 将每个 spadjor 映成殷集，

$$\rho\left(\mathcal{J}_k\right) := \bigcap_{\gamma_i \in \mathcal{J}_k} int\left(\gamma_i\right)$$

对于一个可实现的 spadjor$\mathcal{J}$，令 V 为包含 $\mathcal{J}$ 中约旦曲线之间的所有交点的点集。切割映射 S_V 定义如下：将 $\mathcal{J}$ 中的有向约旦曲线按照 V 中的点分割成一条一条有向道路，将这些道路存放在分段可实现 spadjor$E$ 中。

引理 3.2：黏合映射记为 S_V^-1。给定分段可实现 spadjor E，可实现 spadjor $J = S^-1_V(E)$ 可按如下被唯一构造：

（1）去除 E 中所有的自环（道路首尾相连），并把它们插入 $\mathcal{J}$

（2）从 E 中一条道路 β_{in} 开始，并记 $v \in V$ 为它的末端，如果只存在一条道路 β_{out}，它的开始点为 v，就把它连接到 β_{in} 上。否则，令 β_{out} 为所有以 v 为开始点的道路中，使得 $\angle\beta_{\text{out}}\, v\beta_{\text{in}}$ 最小的道路。重复上述操作直到它变成一个环 γ_1，把构成它的所有道路从 E 中移除

（3）如果 γ_1 是一条约旦曲线，就把它添加到 $\mathcal{J}$，否则把它分成若干约旦曲线后添加到 $\mathcal{J}$。

（4）重复（2）（3）直到 E 为空集。

定义 3.3：补运算 $' : \mathbb{J} \to \mathbb{J}$ 被定义为：

$$\mathcal{J}' := \begin{cases} \hat{1} & \text{如果} \mathcal{J} = \hat{0} \\ \hat{0} & \text{如果} \mathcal{J} = \hat{1} \\ \left(S_V^{-1} \circ R \circ S_V\right)\mathcal{J} & \text{其它} \end{cases}$$

其中 V 是 $\mathcal{J}$ 中约旦曲线的自交点，映射 R 将分段可实现 spadjor $S_V(\mathcal{J})$ 中的每条道路的方向取反。

引理 3.4： 定义 3.3 中的补运算满足：

$$\forall \mathcal{J} \in \mathbb{J}, \quad \rho\left(\mathcal{J}'\right) = (\rho(\mathcal{J}))^{\perp}.$$

定义 3.5： 两个可实现 spadjors $\mathcal{J}$ 和 $\mathcal{K}$ 的交是一个二元运算 $\wedge : \mathbb{J} \times \mathbb{J} \to \mathbb{J}$，定义为

$$\mathcal{J} \wedge \mathcal{K} = \begin{cases} \hat{0} & \text{如果} \mathcal{K} = \hat{0} \\ \mathcal{J} & \text{如果} \mathcal{K} = \hat{1} \\ S_V^{-1}(E) & \text{其它} \end{cases}$$

其中有向多重图（V，E）可按如下构造：

（1）集合 $\mathcal{I} := P(\mathcal{J}) \cap P(\mathcal{K})$ 可能会包含道路和孤立点。将 V 初始化为空集，在 V 中加入所有的这些道路和孤立点。

（2）将 $\mathcal{J}$ 按照 V 中的点进行切割，得到道路集合 $\{\beta_i\} = S_V(\mathcal{J})$。初始化 E 为空集。

（3）对于每个 β_i，如果除去端点包含在 $\rho(\mathcal{K})$ 中或者如果存在 $\beta_j \subset P(\mathcal{K})$ 使得 $\beta_j = \beta_i$ 且它们有相同的方向，那么就把它添加到 E 中。特别的，如果 β_i 是一条约旦曲线且满足以上两个条件之一，我们把它作为一个自环插入 E。

（4）对于每个 $\beta_j \subset S_V(\mathcal{K})$，如果它除去端点包含在 $\rho(\mathcal{J})$ 中，就把它加到 E 中。

引理 3.6： 定义 3.5 中的交运算满足

$$\forall \mathcal{J}, \mathcal{K} \in \mathbb{J}, \quad \rho(\mathcal{J} \wedge \mathcal{K}) = \rho(\mathcal{J}) \cap \rho(\mathcal{K})$$

定义 3.7： 两个可实现 spadjors$\mathcal{J}$ 和 $\mathcal{K}$ 的并是一个二元运算 $\vee : \mathbb{J} \times \mathbb{J} \to \mathbb{J}$，定义为：

$$\forall \mathcal{J}, \mathcal{K} \in \mathbb{J}, \quad \mathcal{J} \vee \mathcal{K} := (\mathcal{J}' \wedge \mathcal{K}')'$$

定理 3.8： 布尔代数 $\left(\mathbb{J}, \vee, \wedge, ', \hat{0}, \hat{1}\right)$ 和 $\left(\mathbb{Y}, \cup^{\perp\perp}, \cap, \perp, \emptyset, \mathbb{R}^2\right)$ 在边界到内部算子 ρ 的作用下是同构的。

4 殷集上的布尔算法

用户友好的设计：输入是算法中不可获取的部分，而一个对用户友好的输入参数的设计可以增加一个算法的吸引力。在我们的实现中，表示殷集的数据结构设计为对可实现 spadjor 的直接编排，也就是一系列数组，其中每个数组表示一个多边形，而点的遍历方向代表了多边形的方向。为了输入殷集，用户不必明确有界约旦曲线之间的包含关系，我们已经将这些信息的计算封装在算法当中。只要每个输入确实是一个可实现 spadjor，算法就会返回正确的结果。与此同时，用户可以选择输出和输入相同格式的结果，也可以输出关于结果之间包含关系的 Hasse 图。这些设计使得软件接口简单、灵活、用户友好。

鲁棒性我们的算法区别于其他布尔代数算法的地方在于用户对布尔代数运算的不确定性有明确的控制。给定一个小的正实数 ϵ，我们定义两点为同一个点如果它们之间的距离小于 ϵ。将这个定义以及它的隐含意义贯穿整个软件包使得我们的实现非常鲁棒，并且能够提供有效的机制用来处理各种刻画拓扑变化的退化情形。在计算几何领域非常有名的一个事实是两条线段的相交可能会引发不可避免的自不相容性并使得一个程序在运行时中止。这一困难的数学核心是线段在无约束的长度尺度下相交带来的潜在的任意病态性。幸运的是，在数值模拟多相流的背景下，总存在一个长度尺度使得低于这个长度尺度的信息是不需要的。所以这个不确定参数 ϵ 不只为了设备的灵活和方便，更重要的是给出了前面所提的计算几何鲁棒性问题的一个简单的解决方法。

实现：实现布尔运算简化为两个计算几何中已经被很好地研究过的问题：第一个是决定一个点和一个简单多边形的相对位置；另一个是计算两个简单多边形边的所有交点。我们解决这两个问题基于扫描平面的思想，但是我们的解法在各个方面都不同于现有的方法。首先，为了鲁棒的浮点计算，我们强调用不确定性准则来处理重叠的线段。其次，我们修改了传统的算法，使得其更适合我们的问题。比如，判断一个点和一个有 n 条边的多边形之间的位置关系，现有方法需要 O(n) 的时间，对 q 个点重复使用该算法的总时间复杂度为 O(qn)，当点数较多时这个方法不是最优的。基于扫描平面的思想，我们设计的算法的时间复杂度为 O((n+q)log n)。

复杂度：在讨论复杂度时，我们限制在用线性多边形构建可实现 spadjor 的

情况。令 n 表示两个可实现 spadjors 的总点数，令 I 表示它们的相交点数。可以证明我们的交算法的复杂度为 O((n + I) log n)。至于补运算，复杂度为 O(n)。我们的构造可实现 spadjor 的 Hasse 图的算法的最坏情况的复杂度为 O(m(m+l) logl)，其中 m 为多边形个数，l 为这些多边形的最大顶点数。

5 结论

我们已经在多相流中引入了流体建模的问题，类比于 CAD 中的实体建模，并且提出了用殷空间去解决这个问题。殷空间是一个简单高效，有着完备布尔代数的拓扑空间。在这个框架下，殷集的拓扑变化可以被捕捉并自然地处理，而拓扑信息如 Betti 数可以在常数时间内被获取。

四、外文原文

MATHEMATICS OF COMPUTATION
Volume 89, Number 325, September 2020, Pages 2333–2364
https://doi.org/10.1090/mcom/3539
Article electronically published on May 8, 2020

BOOLEAN ALGEBRA OF TWO-DIMENSIONAL CONTINUA WITH ARBITRARILY COMPLEX TOPOLOGY

QINGHAI ZHANG AND ZHIXUAN LI

Abstract. We propose a mathematical model for two-dimensional continua in order to establish a solid theoretical foundation for the study of their complex topology, large geometric deformations, and topological changes such as merging in the context of multiphase flows. Our modeling space, named the Yin space, consists of regular open semianalytic sets with bounded boundaries, and is further equipped with constructive and algebraic definitions of Boolean operations. Major distinguishing features of our model include (a) topological information of fluids such as Betti numbers can be easily extracted in constant time, (b) topological changes of fluids are captured by nonmanifold points on fluid boundaries, and (c) Boolean operations on fluids correctly handle all degenerate cases and apply to arbitrarily complex topologies, yet they are simple and efficient in that they only involve determining the relative position of a point to a Jordan curve and intersecting a number of curve segments. Finally, utilities of (a) and (c) are demonstrated by combining the Yin space with the recent cubic MARS method to track a complex fluid in a single vortex flow.

1. Introduction

Physically meaningful regions in the sense of homogeneous continua are ubiquitous, and their modeling is of fundamental significance in innumerable applications of science and engineering. Traditionally, the modeling of physically meaningful regions is the main subject of a mature research field called solid modeling [29, 35]. In comparison, modeling of fluids have always been avoided in the field of multiphase flows. However, rapid advancements in the science of multiphase flows have been calling for such a model so that complex phenomena such as those involving topological changes of fluids can be rigorously studied.

In this paper, we aim to answer this need by introducing the notion of *fluid modeling* in multiphase flows, analogous to solid modeling in computer-aided design (CAD). We propose a topological space for fluid modeling and further equip this space with natural algebraic structures in order to extract essential topological information and to perform simple and efficient Boolean operations.

In Sections 1.1, 1.2, and 1.3, we motivate different aspects of fluid modeling and review previous efforts and relevant results. We then list in Section 1.4 a number of questions as the more detailed targets of this work.

1.1. **Solid modeling.** What distinguishes solid modeling from similar disciplines such as computer graphics is its emphasis on *physical fidelity*, as evident in its

Received by the editor June 19, 2019, and, in revised form, December 29, 2019.
2010 *Mathematics Subject Classification.* Primary 65D18, 76T99.
This work was supported by a grant (approval #11871429) from the National Natural Science Foundation of China.
The first author is the corresponding author.

underlying mathematical and computational principles. This emphasis is natural: driven by the design, analysis, and manufacture of engineering systems, solid modeling must support the representation, visualization, exchange, interrogation, and creation of physical objects in CAD.

One common approach of solid modeling relies on point-set topology. The classical modeling space proposed by Requicha and colleagues [28, 30, 31] consists of *r-sets*, which are bounded, closed, regular semianalytic sets in Euclidean spaces. The regularity condition captures in solid continua the absence of low-dimensional features such as isolated gaps and points, and the semianalytic condition postulates that the boundary of a solid be locally well behaved; see Section 3.1 for more details.

The other common approach in solid modeling is *combinatorial*, in the sense of *cell complexes* in algebraic topology [22, 34]. Complex objects are viewed in terms of primitive building blocks called *cells*, thus it is not the constituting cells but their combinatorial informations that describe the physical object. Take simplicial complexes for example: a *k-cell* is a k-simplex in the Euclidean space $\mathbb{R}^k$, and many cells of different dimensions are glued together to form an n-dimensional simplicial complex by requiring that any adjacent pair of k-cells be attached to each other along a $(k-1)$-cell for each $k = 1, \ldots, n$. The adjacency of k-cells is encoded in the *kth boundary operator* ∂_k, which maps each k-cell to an element in the $(k-1)$-*chain* C_{k-1}, a group of formal sums of $(k-1)$-cells. If we concatenate the chain groups with the boundary operators, we obtain a *chain complex*,

$$C_n \xrightarrow{\partial_n} \cdots \xrightarrow{\partial_{k+1}} C_k \xrightarrow{\partial_k} \cdots \xrightarrow{\partial_1} C_0 \ , \tag{1.1}$$

where each boundary operator is a group homomorphism and any two adjacent operators concatenate to the zero map. This chain complex is all we need for mathematical modeling and computer representation of any n-dimensional solid! As a prominent advantage, key topological quantities, such as the number of connected components and the number of holes, can be systematically computed from the chain complex. The cost of this computation, however, can be substantial [15].

Thanks to the fact that any r-set can be represented by a simplicial complex as accurately as one wishes, the point-set approach and the combinatorial approach are seamlessly consistent [30]. Thus we can use these two models interchangeably. This consistency also forces an n-dimensional r-set to be closed; otherwise a boundary operator in (1.1) may have a range outside of the chain complex.

1.2. **Fluid modeling and interface tracking (IT) in multiphase flows.** In dramatic comparison to the aforementioned research on solid modeling, efforts on geometric modeling of fluids are rare: mathematical models and computer algorithms have been deliberately designed such that *geometric* modeling of fluids is avoided in numerically simulating multiphase flows.

In the volume-of-fluid (VOF) method [12], a deforming fluid phase M is represented by a *color function* $f(\mathbf{x}, t)$,

$$f(\mathbf{x}, t) := \begin{cases} 1 & \text{if there is } \mathsf{M} \text{ at } (\mathbf{x}, t); \\ 0 & \text{otherwise;} \end{cases} \tag{1.2}$$

then the region occupied by M at time t is a point set

$$\mathcal{M}(t) := \{\mathbf{x} : f(\mathbf{x}, t) = 1\}. \tag{1.3}$$

Either the scalar conservation law

$$\frac{\partial f}{\partial t} + \nabla \cdot (f\mathbf{u}) = 0 \tag{1.4}$$

or the advection equation

$$\frac{\partial f}{\partial t} + \mathbf{u} \cdot \nabla f = 0 \tag{1.5}$$

is solved to recover the boundary of $\mathcal{M}$ at subsequent time instants. In the level-set method [25], the boundary of a fluid phase is represented as the zero isocontour of a signed distance function ϕ, and once again the region of the fluid phase is recovered by numerically solving either (1.4) or (1.5) on ϕ. In the front tracking method [40], the boundary of a fluid phase is represented by connected Lagrangian markers; tracking the fluid phase is then reduced to tracking these markers via numerically solving ordinary differential equations. In all of these IT methods, geometric problems in deforming fluids with sharp interfaces are converted to numerically solving differential equations. With topological information discarded, this conversion largely reduces the complexity of IT both theoretically and computationally; this is a main reason for successes of the aforementioned IT methods. During the past forty years, these IT methods have been extremely valuable in studying multiphase flows.

As the science of multiphase flows moves towards more and more complex phenomena, higher and higher expectations are imposed on IT. First, the wider and wider spectrum of relevant time scales and length scales in mainstream problems demands that IT methods be more and more accurate and efficient. Second, the tight coupling of interface to ambient fluids necessitates accurate estimation of derived geometric quantities such as curvature and unit normal/tangential vectors. Third, topological changes of a fluid phase such as merging exhibit distinct behaviors for different regimes of the Weber number and other impact parameters [27], hence it is not enough to handle topological changes solely from the interface locus and the velocity field. For these problems, an IT method should also take as its input a policy that describes how the interface shall evolve at the branching time and place of topological changes.

Despite their tremendous successes, current IT methods have a number of limitations in answering the aforementioned challenges of multiphase flows. First, most methods are at best second-order accurate [43]. Second, the IT errors put an upper limit on the accuracy of estimating curvature and unit vectors. It is shown in [47] that, for a second-order method, its error of curvature estimation is proportional to $\sqrt{\epsilon_p}$, where ϵ_p denotes a norm of IT errors. In other words, the number of accurate digits one gets in curvature estimation is at best half of that in the IT results. Third, the avoidance of geometric modeling of fluids renders it highly difficult to treat topological changes rigorously. When merging and separation happen, front-tracking methods have to resort to "surgical" operations that are short of theoretical justification. VOF methods and level-set methods have no special procedures for topological changes; this is often advertised as an advantage. However, by using this "automatic" treatment, an application scientist has *no control* over the evolution of an interface that undergoes topological changes: the evolution is determined not by the physics, but by particularities of numerical algorithms [43, 48]. Clearly, this disadvantage is a consequence of avoiding the geometric and topological modeling of fluids.

In this work, we aim to *establish a theoretical foundation for fluid modeling*. To prevent reinventing the wheel, we have tried to utilize the wealth of solid modeling, but found that none of the two main approaches in Section 1.1 is adequate for fluid modeling. Topology computing based on cell complexes involves much machinery, yet its efficiency may not be acceptable; nor are the r-sets suitable for fluid modeling. First, the requirement of r-sets being closed is not amenable to numerical analysis of IT methods [43]. Second, topological changes create on the fluid boundary a special type of nonmanifold points, such as the q points in Figure 1, which cause nonuniqueness of boundary representations; cf. Figure 5. This nonuniqueness degrades isomorphisms to homomorphisms in the association of algebraic structures with elements in the topological modeling space. Third, r-sets can be modified to yield a new Boolean algebra whose implementation is much simpler and more efficient.

1.3. **Boolean operations.** The operations to be performed on the modeling space are an indispensable part of the modeling process; after all, a major purpose of modeling is to answer questions on the objects being modeled. Hence theoretically a modeling space is shaped by primary cases of queries. Computationally, these operations should be defined algebraically and constructively so that they furnish realizable algorithms that finish in reasonable time.

We are interested in Boolean operations on physically meaningful regions with arbitrarily complex topologies. This interest follows naturally from the motivations in Section 1.2. First, we have shown that algorithms for clipping splinegons with a linear polygon can improve the IT accuracy by many orders of magnitudes [48]. Second, for coupling an IT method to an Eulerian main flow solver, regions occupied by the fluid inside fixed control volumes are needed to define averaged values and to construct stencils for approximating spatial operators with linear combinations of these averaged values. Third, in handling topological changes, the emerging time and sites of nonmanifold points need to be detected on the fluid boundary before we are able to decide how to evolve it. This detecting problem requires calculating intersections of multiple regions inside a single control volume.

Boolean operations on polygons are an active and intense research topic in many related fields such as computational geometry, computer graphics, CAD, and the geographic information system (GIS). In particular, physically meaningful regions in GIS such as parks, roads, and lakes are represented by polygons, and their Boolean operations are essential for extracting information and answering queries. Consequently, there exist numerous papers on this topic; see, e.g., [7, 10, 18, 19, 21, 24, 36, 38, 41] and the references therein. However, many current algorithms are subject to strong restrictions on operand polygons such as convexity, simple-connectedness, and no self-intersections. In addition, most algorithms fail for degenerate cases such as a vertex of a polygon being on the edge of the other polygon; these degenerate scenarios, nonetheless, are at the core of characterizing and treating topological changes. Therefore, current Boolean algorithms are not suitable for fluid modeling.

As another main reason for their lack of applicability in multiphase flows, very few of current Boolean algorithms have a solid mathematical foundation, and those that do have other notable drawbacks. For example, Boolean algorithms based on cell complexes [26, 32] seem to be inefficient for complex topologies. Those based on Nef polyhedra [3, 11, 23] have an elegant theoretical foundation and are

applicable to arbitrarily complex topologies, but they appear as an overkill for fluid modeling in that many elements in the modeling space of Nef polyhedra do not have counterparts in multiphase flows. In addition, the corresponding algorithms and data structures are complicated and difficult to implement. For both types of algorithms, computing the topological information such as Betti numbers would be very time-consuming.

1.4. **Motivations and contributions of this work.** Methods that couple elementary concepts or tools from multiple disciplines often perform surprisingly well. For fluid-structure interactions, a recent approach called isogeometric analysis [1, 6] has been increasingly popular, and much of its success is due to the integration of finite element methods with highly accurate (and sometimes exact) solid modeling in CAD. In our previous work, we have adopted a similar guiding principle to integrate IT with a topological space for fluid modeling [49]. The resulting generic framework, called MARS, furnishes new tools for analyzing current IT methods [43] and leads to a new IT method and a new curvature-estimation algorithm that are more accurate than current methods by many orders of magnitudes [47, 48].

In recognition of the potentially large benefits of integrating fluid modeling with multiphase flows and in view of the discussions in previous subsections, we list a number of questions as the driving forces behind this work.

(Q-1) Can we propose a generic topological space that appropriately models physically meaningful regions across multiple research fields such as solid modeling, GIS, and multiphase flows?
(Q-2) Can we find a simple representation scheme for elements in the modeling space to facilitate geometric and topological queries?
(Q-3) Can we design simple and efficient Boolean operations that correctly handle all degenerate cases?
(Q-4) In particular, can we provide theoretical underpinning and algorithmic support for handling topological changes of moving regions?
(Q-5) Meanwhile, can we extract topological information such as Betti numbers with optimal complexity?

In this paper, we provide positive answers to all of the above questions. (Q-1) is answered in Section 3, where physically meaningful regions are modeled by a topological space, called the Yin space, which consists of regular open semianalytic sets with bounded boundaries. These conditions capture fluid features that are commonly relevant in multiphase flows, solid modeling, and GIS. Furthermore, Yin sets are defined in terms of computable *mathematical* properties and are thus independent of any particular representation or specific application. As such, this modeling space serves as a bridge between multiphase flows and the other fields that emphasize geometry and topology. As our answer to (Q-2), each Yin set can be uniquely expressed as the result of finite Boolean operations on interiors of oriented Jordan curves. This uniqueness leads to an isomorphism from the Yin space into the Jordan space, a collection of certain posets of oriented Jordan curves, and this isomorphism reduces Boolean algebra on the two-dimensional Yin space to one-dimensional routines in the Jordan space, namely locating a point relative to a simple polygon and finding intersections of curve segments. This is our answer to (Q-3); see Section 4.

In addressing (Q-4), we pay special attention to issues related to nonmanifold points on the fluid boundary, such as characterizing topological changes with improper intersections of curves and dividing closed curves at these improper intersections to ensure correctness of Boolean operations. However, we emphasize that, in both our theory and our algorithms, nonmanifold points of topological changes are treated not as an anomaly, but as a natural consequence of capturing the physical meaningfulness of fluids with the mathematical conditions that constitute the notion of Yin sets. This is a major advantage of the Yin sets over the r-sets.

As another prominent feature of our theory, the number of connected components in any bounded Yin set is simply the number of positively oriented Jordan curves in its boundary representation, and the number of holes in a component is the number of negatively oriented Jordan curves in the boundary representation of that component. Since these numbers are returned in $O(1)$ time, our answer to (Q-5) is of optimal complexity.

The rest of this paper is organized as follows. In Section 2, we introduce prerequisites and notation. In Section 3, we propose Yin sets as our fluid modeling space and study its topological properties. In Section 4, we design Boolean operations on the Yin space in a way so that corresponding algorithms can be implemented by straightforward orchestration of these definitions. In Section 5, we give algorithmic details on implementing the Boolean algebra of Yin sets. Utilizing the Bentley–Ottmann paradigm of plane sweeping [2] in calculating intersections of curve segments, our implementation of the Boolean operations is close to optimal complexity. A number of fun tests are given in Figure 13 to illustrate the Boolean algorithm. In Section 5.5, we approach interface tracking of multiphase flows from the viewpoint of the actions of continuous flow maps upon Yin sets. Results of numerical tests demonstrate that interface tracking based on the proposed Boolean algorithms preserves complex topologies of the fluid if the flow map is homeomorphic. Finally, we draw the conclusion and discuss several research prospects in Section 6.

2. Preliminaries

In this section we collect relevant definitions and theorems that form the *algebraic* foundation of this work. Some notation introduced here will be repeatedly used in subsequent sections.

2.1. **Partially ordered sets.** The Cartesian product of a nonempty set $\mathcal{A}$ with itself n times is denoted by $\mathcal{A}^n$; in particular, $\mathcal{A}^0 = \{\emptyset\}$. An *$n$-ary relation on $\mathcal{A}$* is a subset of $\mathcal{A}^n$; if $n = 2$ it is called a *binary relation*. A given binary relation "$\sim$" on a set $\mathcal{A}$ is said to be an *equivalence relation* if and only if it is *reflexive* ($a \sim a$), *symmetric* ($a \sim b \ \Rightarrow \ b \sim a$), and *transitive* ($a \sim b, b \sim c \ \Rightarrow \ a \sim c$) for all $a, b, c \in \mathcal{A}$. A binary relation "$\leq$" defined on a set $\mathcal{A}$ is a *partial order* on $\mathcal{A}$ if and only if it is reflexive ($a \leq a$), *antisymmetric* ($a \leq b, b \leq a \ \Rightarrow \ b = a$), and transitive ($a \leq b, b \leq c \ \Rightarrow \ a \leq c$) for all $a, b, c \in \mathcal{A}$.

A nonempty set $\mathcal{A}$ with a partial order $\leq$ on it is called a *partially ordered set*, or more briefly a *poset*. Two elements $a, b \in \mathcal{A}$ are *comparable* if either $a \leq b$ or $b \leq a$; otherwise a and b are *incomparable*. If all $a, b \in \mathcal{A}$ are comparable by $\leq$, then "$\leq$" is a *total order* on $\mathcal{A}$ and $\mathcal{A}$ is a *chain* or linearly ordered set. For examples, $\mathbb{R}$ with the usual order of real numbers is a chain; the *power set* of $\mathcal{A}$, i.e., the set of all

subsets of $\mathcal{A}$, with the subset relation "$\subseteq$" is a poset but not a chain. The notation $a \geq b$ means $b \leq a$, and $a < b$ means both $a \leq b$ and $a \neq b$.

Definition 2.1 (Covering relation)**.** Let $\mathcal{A}$ denote a poset, and let $a, b \in \mathcal{A}$. We say b *covers* a and write $a \prec b$ or $b \succ a$ if and only if $a < b$ and no element $c \in \mathcal{A}$ satisfies $a < c < b$.

Most concepts on the ordering of $\mathbb{R}$ make sense for posets. Let $\mathcal{A}$ be a subset of a poset $\mathcal{P}$. An element $p \in \mathcal{P}$ is an *upper bound* of $\mathcal{A}$ if $a \leq p$ for all $a \in \mathcal{A}$. $p \in \mathcal{P}$ is the *least upper bound* of $\mathcal{A}$, or *supremum* of $\mathcal{A}$ ($\sup \mathcal{A}$) if p is an upper bound of $\mathcal{A}$, and $p \leq b$ for any upper bound b of $\mathcal{A}$. Similarly, we can define the concepts of a *lower bound* and the *greatest lower bound* of $\mathcal{A}$ or the *infimum* of $\mathcal{A}$ ($\inf \mathcal{A}$).

Definition 2.2 (Lattice as a poset)**.** A *lattice* is a poset $\mathcal{L}$ satisfying that, for all $a, b \in \mathcal{L}$, both $\sup\{a, b\}$ and $\inf\{a, b\}$ exist in $\mathcal{L}$.

2.2. **Distributive lattices.** An *n-ary operation on* $\mathcal{A}$ is a function $f : \mathcal{A}^n \to \mathcal{A}$ where n is the *arity* of f. A *finitary* operation f is an n-ary operation for some nonnegative integer $n \in \mathbb{N}$. f is nullary (or a constant) if its arity is zero, i.e., it is completely determined by the only element $\emptyset \in \mathcal{A}^0$, and hence a nullary operation f on $\mathcal{A}$ can be identified with the element $f(\emptyset)$; for convenience it is regarded as an element of $\mathcal{A}$. An operation on $\mathcal{A}$ is *unary* or *binary* if its arity is 1 or 2, respectively.

Definition 2.3 (Universal algebra)**.** An *algebra* is an ordered pair $\mathbf{A} := (\mathcal{A}, \mathcal{F})$ where $\mathcal{A}$ is a nonempty set and $\mathcal{F}$ a family of finitary operations on $\mathcal{A}$. The set $\mathcal{A}$ is the *universe* or the *underlying set* of $\mathbf{A}$ and $\mathcal{F}$ the *fundamental operations* of $\mathbf{A}$.

An algebra is *finite* if the cardinality of its universe is bounded. When $\mathcal{F}$ is finite, say $\mathcal{F} = \{f_1, f_2, \ldots, f_k\}$, we also write $\mathbf{A} = (\mathcal{A}, f_1, f_2, \ldots, f_k)$ with the operations sorted by their arities in descending order. As a common example, a *group* is an algebra of the form $\mathbf{G} = (\mathcal{G}, \cdot, ^{-1}, 1)$ where $\cdot, ^{-1}, 1$ are a binary, a unary, and a nullary operation on $\mathcal{G}$, respectively.

Definition 2.4 (Lattice as an algebra)**.** A *lattice* is an algebra $\mathbf{L} := (\mathcal{L}, \mathcal{F})$, where $\mathcal{F}$ contains two binary operations $\vee$ and $\wedge$ (read "join" and "meet", respectively) on $\mathcal{L}$ that satisfy the following axiomatic identities for all $x, y, z \in \mathcal{L}$:

(LA-1) commutative laws: $x \vee y = y \vee x$, $x \wedge y = y \wedge x$;
(LA-2) associative laws: $x \vee (y \vee z) = (x \vee y) \vee z$, $x \wedge (y \wedge z) = (x \wedge y) \wedge z$;
(LA-3) absorption laws: $x = x \vee (x \wedge y)$, $x = x \wedge (x \vee y)$.

Sometimes the following idempotent laws are also included in the definition of a lattice although they can be derived from the above three axioms:

$$x \vee x = x, \qquad x \wedge x = x. \tag{2.1}$$

A lattice defined as a poset can be converted to an algebra by constructing the binary operations as $a \vee b = \sup\{a, b\}$ and $a \wedge b = \inf\{a, b\}$; the converse case can also be achieved by defining the partial order as $a \leq b \Leftrightarrow a = a \wedge b$. Hence Definitions 2.2 and 2.4 are equivalent.

Definition 2.5. A *bounded lattice* is an algebra $(\mathcal{L}, \mathcal{F})$, where $\mathcal{F}$ contains binary operations $\vee, \wedge$ and nullary operations $\hat{0}, \hat{1}$ so that $(\mathcal{L}, \vee, \wedge)$ is a lattice and $\forall x \in \mathcal{L}$,

$$x \wedge \hat{0} = \hat{0}, \qquad x \vee \hat{1} = \hat{1}. \tag{2.2}$$

The boundedness in the above definition is best understood from the poset viewpoint: $\hat{0} \leq x$ and $x \leq \hat{1}$ for all $x \in \mathcal{L}$.

Definition 2.6. A *distributive lattice* is a lattice which satisfies either of the distributive laws

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z), \qquad x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z). \tag{2.3}$$

Either identity in (2.3) can be deduced from the other and from Definition 2.4 [5, p. 10].

A notion central to every branch of mathematics is an isomorphism. In particular, two lattices are isomorphic if they have the same structure.

Definition 2.7 (Lattice isomorphism)**.** A *homomorphism* of the lattice $(\mathcal{L}_1, \vee, \wedge)$ into the lattice $(\mathcal{L}_2, \cup, \cap)$ is a map $\phi : \mathcal{L}_1 \to \mathcal{L}_2$ satisfying

$$\forall x, y \in \mathcal{L}_1, \quad \phi(x \vee y) = \phi(x) \cup \phi(y), \ \phi(x \wedge y) = \phi(x) \cap \phi(y). \tag{2.4}$$

An *isomorphism* is a bijective homomorphism.

More details on distributive lattices can be found in [5] from the perspective of universal algebra and in [37, ch. 3] from the viewpoint of posets. See [9] for a more accessible one.

2.3. **Boolean algebra.** The simplest definition may be due to Huntington [13].

Definition 2.8. A *Boolean algebra* is an algebra of the form

$$\mathbf{B} := (\mathcal{B}, \ \vee, \ \wedge, \ ', \ \hat{0}, \ \hat{1}), \tag{2.5}$$

where the binary operations $\vee$, $\wedge$, the unary operation $'$ called a complementation, and the nullary operations $\hat{0}$, $\hat{1}$ satisfy

(BA-1) the identity laws $x \wedge \hat{1} = x$, $x \vee \hat{0} = x$,
(BA-2) the complement laws $x \wedge x' = \hat{0}$, $x \vee x' = \hat{1}$,
(BA-3) the commutative laws (LA-1),
(BA-4) the distributive laws (2.3).

Other definitions contain redundant axiomatic laws that can be deduced from the above four conditions. For example, Givant and Halmos [8, p. 10] defined a Boolean algebra as an algebra with its fundamental operations satisfying (LA-1), (LA-2), (2.1), (2.2), (2.3), (BA-1), (BA-2), $\hat{0}' = \hat{1}$, $\hat{1}' = \hat{0}$, $(x')' = x$, and the DeMorgan laws,

$$(x \wedge y)' = x' \vee y', \qquad (x \vee y)' = x' \wedge y'. \tag{2.6}$$

In this work we adopt the viewpoint of Burris and Sankappanavar [5, p. 116].

Definition 2.9. A *Boolean algebra* is a bounded distributive lattice with an additional complementation operation that satisfies the complement laws (BA-2).

2.4. **Veblen's theorem.** A *graph* is an ordered pair $G = (V, E)$, where V is the set of *vertices* and E the set of *edges*, with each edge being an *unordered* pair of distinct vertices. G' is a *subgraph* of G, written as $G' \subseteq G$, if $V(G') \subseteq V(G)$ and $E(G') \subseteq E(G)$. If $uv \in E(G)$, then u and v are *adjacent* in G, and the edge uv is said to be *incident* to u and v. The *degree* of a vertex v is the number of edges incident to v.

Definition 2.10. A *path* is a graph P of the form $V(P) = \{v_0, v_1, \ldots, v_\ell\}$ and $E(P) = \{v_0v_1, v_1v_2, \ldots, v_{\ell-1}v_\ell\}$. A *cycle* is a graph of the form $C := P + v_0v_\ell$, where P is a path and $\ell \geq 2$.

A graph G is *connected* if for every pair of distinct vertices in $V(G)$ there is a subgraph of G as the path from one vertex to the other. A *component* of the graph is a *maximal connected subgraph*.

Theorem 2.11 (Veblen [42])**.** *A graph can be partitioned into edge-disjoint cycles if and only if the degree of every vertex is even.*

Proof. If a graph is the union of a number of edge-disjoint cycles, then clearly a vertex contained in n cycles has degree $2n$. Hence the necessity holds.

Suppose that the degree of every vertex is a positive even integer. How do we find a single cycle in G? Let $P = x_0x_1 \cdots x_\ell$ be a path of maximal length ℓ in G. Since $d(x_0) \geq 2$, x_0 must have another neighbor y in addition to x_1. Furthermore, we must have $y = x_i$ for some $i \in [2, \ell]$; otherwise it would contradict the starting condition that P is of maximal length. Therefore we have found a cycle $x_0x_1 \cdots x_i$.

Having found one cycle, we remove it from G. If the remaining subgraph G_1 of G is not empty, then the degree of every vertex in G_1 remains positive and even. Repeating the cycle-finding procedures completes the proof; see [4, p. 5]. □

A *multigraph* is an augmented graph that allows *loop*s and *multiple edges*; the former is defined as a special edge joining a vertex to itself and the latter several edges joining the same vertices. A loop contributes 2 to the degree of a vertex, while each edge in a multiple edge contributes 1. As for cycles, the condition of $\ell \geq 2$ in Definition 2.10 is changed to $\ell \geq 0$ for a multigraph: $\ell = 0$ indicates a loop and $\ell = 1$ two edges joining the same vertices. It is straightforward to extend Theorem 2.11 to multigraphs.

Theorem 2.12. *A multigraph can be partitioned into edge-disjoint cycles if and only if the degree of each vertex is positive and even.*

A *directed* graph/multigraph is a graph/multigraph where the edges are *ordered* pairs of vertices. An edge uv is then said to *start* at u and *end* at v. Furthermore, the degree of a vertex v is split into the *outdegree* $d^+(v)$ and the *indegree* $d^-(v)$, with the former as the number of edges starting at v and the latter that of edges ending at v. Veblen's Theorem generalizes to directed multigraphs in a straightforward manner.

Theorem 2.13. *A directed multigraph can be partitioned into directed cycles if and only if each vertex has the same outdegree and indegree.*

3. Yin sets

Based on regular semianalytic sets introduced in Section 3.1, we propose in Section 3.2 the Yin space for fluid modeling in two dimensions. From the viewpoint of Jordan curves in Section 3.3, we study in Section 3.4 the local and global topology of Yin sets, the results of which yield the notion of *realizable spadjors* in Section 3.5 as a unique boundary representation of Yin sets.

3.1. **Regular semianalytic sets.** In a topological space $\mathcal{X}$, the *complement* of a subset $\mathcal{P} \subseteq \mathcal{X}$, written as $\mathcal{P}'$, is the set $\mathcal{X} \setminus \mathcal{P}$. The *closure* of a set $\mathcal{P} \subseteq \mathcal{X}$, written as $\mathcal{P}^-$, is the intersection of all closed supersets of $\mathcal{P}$. The *interior* of $\mathcal{P}$, written as $\mathcal{P}^\circ$, is the union of all open subsets of $\mathcal{P}$. The *exterior* of $\mathcal{P}$, as written as $\mathcal{P}^\perp := \mathcal{P}'^\circ := (\mathcal{P}')^\circ$, is the interior of its complement. By the identity $\mathcal{P}^- = \mathcal{P}'^{\circ\prime}$ [8, p. 58], we have $\mathcal{P}^\perp = \mathcal{P}^{-\prime}$. A point $\mathbf{x} \in \mathcal{X}$ is a *boundary point* of $\mathcal{P}$ if $\mathbf{x} \notin \mathcal{P}^\circ$ and $\mathbf{x} \notin \mathcal{P}^\perp$. The *boundary* of $\mathcal{P}$, written as $\partial\mathcal{P}$, is the set of all boundary points of $\mathcal{P}$. It can be shown that $\mathcal{P}^\circ = \mathcal{P} \setminus \partial\mathcal{P}$ and $\mathcal{P}^- = \mathcal{P} \cup \partial\mathcal{P}$. An open set $\mathcal{P} \subseteq \mathcal{X}$ is *regular* if it coincides with the interior of its own closure, i.e., if $\mathcal{P} = \mathcal{P}^{-\circ}$. A closed set $\mathcal{P} \subseteq \mathcal{X}$ is *regular* if it coincides with the closure of its own interior, i.e., if $\mathcal{P} = \mathcal{P}^{\circ-}$. The duality of the interior and closure operators implies $\mathcal{P}^\circ = \mathcal{P}'^{-\prime}$, hence $\mathcal{P}$ is a *regular open set* if and only if $\mathcal{P} = \mathcal{P}^{\perp\perp} := \left(\mathcal{P}^\perp\right)^\perp$. For any subset $Q \subseteq \mathcal{X}$, it can be shown that $Q^{\perp\perp}$ is a regular open set and $Q^{\circ-}$ is a regular closed set.

Regular sets, open or closed, capture the salient feature that physically meaningful regions are free of lower-dimensional elements such as isolated points and curves in 2D and dangling faces in 3D.

Theorem 3.1 (MacNeille [20] and Tarski [39])**.** *Let $\mathbb{B}$ denote the class of all regular open sets of a topological space $\mathcal{X}$ and define $\mathcal{P} \cup^{\perp\perp} \mathcal{Q} := (\mathcal{P} \cup \mathcal{Q})^{\perp\perp}$ for all $\mathcal{P}, \mathcal{Q} \subseteq \mathcal{X}$. Then $\mathbf{B}_o := (\mathbb{B},\ \cup^{\perp\perp},\ \cap,\ ^\perp,\ \emptyset,\ \mathcal{X})$ is a Boolean algebra.*

Proof. See [8, §10]. □

Similarly, it can be shown that, with appropriately defined operations, regular closed sets of a topological space $\mathcal{X}$ also form a Boolean algebra [17, p. 39].

Regular sets are not perfect for representing physically meaningful regions yet; some of them cannot be described by a finite number of symbol structures. For example, some sets have nowhere differentiable boundaries, which, in their parametric forms, are usually infinite series of continuous functions [33]. Another pathological case is more subtle: intersecting two regular sets with piecewise smooth boundaries may yield an infinite number of disjoint regular sets. Consider

$$\begin{cases} \mathcal{A}_p := \{(x, y) \in \mathbb{R}^2 : -2 < y < \sin\frac{1}{x},\ 0 < x < 1\}, \\ \mathcal{A}_s := \{(x, y) \in \mathbb{R}^2 : 0 < y < 1,\ -1 < x < 1\}. \end{cases} \tag{3.1}$$

Although both $\mathcal{A}_p$ and $\mathcal{A}_s$ are described by two inequalities, their intersection is a disjoint union of an infinite number of regular sets; see [28, Fig. 4-1, Fig. 4-2]. This poses a fundamental problem where results of Boolean operations of two regular sets may not be well represented on a computer by a finite number of entitites.

Therefore, we need to find a proper subspace of regular sets, each element of which is finitely describable. This search eventually arrives at semianalytic sets.

Definition 3.2. A set $\mathcal{S} \subseteq \mathbb{R}^{\mathsf{D}}$ is *semianalytic* if there exist a finite number of analytic functions $g_i : \mathbb{R}^{\mathsf{D}} \to \mathbb{R}$ such that $\mathcal{S}$ is in the universe of a finite Boolean algebra formed from the sets

$$\mathcal{X}_i = \left\{\mathbf{x} \in \mathbb{R}^{\mathsf{D}} : g_i(\mathbf{x}) \geq 0\right\}. \tag{3.2}$$

The g_i's are called the *generating functions* of $\mathcal{S}$. In particular, a semianalytic set is *semialgebraic* if all of its generating functions are polynomials.

(a) an unbounded Yin set (b) a bounded Yin set

FIGURE 1. Examples of Yin sets. The Yin set in (a) is obtained by removing from $\mathbb{R}^2$ three closed balls, two of which share a common boundary point q. The Yin set in (b) is the union of four pairwise disjoint Yin sets $\mathcal{Y} = \bigcup_{i=1}^{4} \mathcal{Y}_i$, where $\mathcal{Y}_4$ is an open ellipse with three closed balls removed, two of which share a common boundary point q_2. The points q, q_1, q_2 are boundary points but not interior points of the Yet sets.

Recall that a function is *analytic* if and only if its Taylor series at $\mathbf{x}_0$ converges to the function in some neighborhood for every $\mathbf{x}_0$ in its domain. In the example of (3.1), $\mathcal{A}_s$ is semianalytic while $\mathcal{A}_p$ is not, because the Taylor series of $g_2(x, y) := \sin \frac{1}{x} - y$ at the origin does not converge. Roughly speaking, the boundary curves of regular semianalytic sets are piecewise smooth.

3.2. $\mathbb{Y}$**: The Yin space for fluid modeling.** Regular closed semianalytic sets have been an essential mathematical tool for solid modeling since the dawning time of this field [31]. However, as shown in Figure 5, requiring regular semianalytic sets to be closed would make their boundary representation not unique, degrading the isomorphism in Definition 3.20 and Theorem 3.21 to a homomorphism. As another work closely related to this one, the analysis on a family of interface tracking methods [43, 49] via the theory of donating regions [44, 45] also requires that the regular sets be open. Therefore, only regular open semianalytic sets are employed in this work.

Definition 3.3. A *Yin set*[1] $\mathcal{Y} \subseteq \mathbb{R}^2$ is a regular open semianalytic set whose boundary is bounded. The class of all such Yin sets form the *Yin space* $\mathbb{Y}$.

In Figure 1, the Yin set in subplot (a) is unbounded and connected while that in subplot (b) is bounded and consists of four disjoint components.

By Definition 3.2, semianalytic sets are closed under set complementation, finite union, and finite intersection. Then by Theorem 3.1, regular open semianalytic sets form a Boolean algebra since they are the intersection of the universes of two Boolean algebras. Furthermore, Boolean operations on Yin sets preserve the

[1]Yin sets are named after the first author's mentor, Madam Ping Yin. As a coincidence, the most important dichotomy in Taoism consists of Yin and Yang, where Yang represents the active, the straight, the ascending, and so on, while Yin represents the passive, the circular, the descending, and so on. From this viewpoint, straight lines and Jordan curves can be considered as Yang 1-manifolds and Yin 1-manifolds, respectively.

attribute of a bounded boundary being bounded, and hence we have the following theorem.

Theorem 3.4. *The algebra* $\mathbf{Y} := (\mathbb{Y},\ \cup^{\perp\perp},\ \cap,\ ^{\perp},\ \emptyset,\ \mathbb{R}^2)$ *is a Boolean algebra.*

3.3. **Jordan curves and orientations.** A *path* in $\mathbb{R}^2$ from p to q is a continuous map $f : [0,1] \to \mathbb{R}^2$ satisfying $f(0) = p$ and $f(1) = q$. A subset $\mathcal{P}$ of $\mathbb{R}^2$ is *path-connected* if every pair of points of $\mathcal{P}$ can be joined by a path in $\mathcal{P}$. Given $\mathcal{Q} \subset \mathbb{R}^2$, define an equivalence relation on $\mathcal{Q}$ by setting $x \sim y$ if there is a path-connected subset of Q that contains both x and y. The equivalence classes are called the *path-connected component*s of $\mathcal{Q}$.

A *planar curve* is a continuous map $\gamma : (0,1) \to \mathbb{R}^2$. It is *piecewise analytic* if the map is the composite of a finite number of analytic functions. It is *simple* if the map is injective; otherwise it is *self-intersecting*. Although strictly speaking a curve γ is a map, we also use γ to refer to its image. Two distinct curves γ_1 and γ_2 *intersect* at q if there exist $s_1, s_2 \in (0,1)$ such that $\gamma_1(s_1) = \gamma_2(s_2) = q$. Then q is the *intersection* of γ_1 and γ_2. For an open ball $\mathcal{N}_r(q)$ with sufficiently small radius r, $\mathcal{N}_r(q) \setminus \gamma_1$ consists of two disjoint connected regular open sets. If $\gamma_2 \setminus q$ is entirely contained in one of these two sets, q is an *improper intersection*; otherwise it is a *proper intersection*. Two curves are *disjoint* if they have neither proper intersections nor improper ones. Suppose upon its extension to a path, a simple curve γ further satisfies $\gamma(0) = \gamma(1)$; then γ is a *simple closed curve* or a *Jordan curve*.

Theorem 3.5 (Jordan Curve Theorem [14])**.** *The complement of a Jordan curve* γ *in the plane* $\mathbb{R}^2$ *consists of two components, each of which has* γ *as its boundary. One component is bounded and the other is unbounded; both of them are open and path-connected.*

The above theorem states that a Jordan curve divides the plane into three parts: itself, its *interior*, and its *exterior*.

Definition 3.6. The *interior* of an oriented Jordan curve γ, denoted by $\mathrm{int}(\gamma)$, is the component of the complement of γ that always lies to the left when an observer traverses the curve in the increasing direction of the parameterization s.

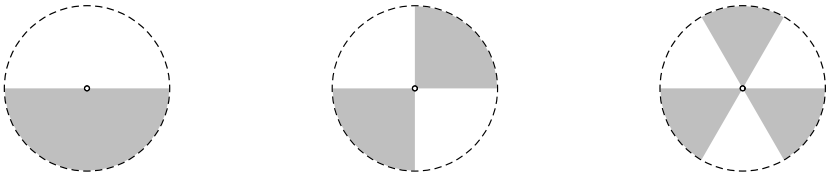
A Jordan curve is said to be *positively oriented* if its interior is the bounded component of its complement; otherwise it is *negatively oriented*. The orientation of a Jordan curve can be flipped by reversing the increasing direction of the parameterization. The following notion will be used throughout this work.

Definition 3.7. Two Jordan curves are *almost disjoint* if they have no proper intersections and at most a finite number of improper intersections.

3.4. **The local and global topology of a Yin set.** The following lemma characterizes the local topology of a Yin set at its boundary.

Lemma 3.8. *Let* $p \in \partial\mathcal{Y}$ *be a boundary point of a Yin set* $\mathcal{Y} \subset \mathbb{R}^2$*, and denote by* $\mathcal{N}_r(p)$ *the open ball centered at* p *with its radius* $r > 0$*. For any sufficiently small* r*,*

(a) $\partial\mathcal{Y} \cap \mathcal{N}_r(p)$ *consists of* $n_c(p)$ *simple curves, where* $n_c(p)$ *is a finite positive integer,*
(b) *if* $n_c(p) > 1$*, all simple curves in* (a) *intersect and* p *is their sole intersection,*



(a) one curve (b) two curves (c) three or more

FIGURE 2. The local topology at a boundary point p (open dot) of a Yin set $\mathcal{Y}$ (shaded region). The dashed circle represents the boundary of $\mathcal{N}_r(p)$, a local neighborhood of p. The open dot in subplot (c) corresponds to the boundary point q_2 in Figure 1(b), while that in subplot (b) to q, q_1 in Figure 1(a), (b). All other boundary points in Figure 1 correspond to those in subplot (a).

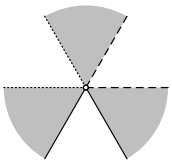


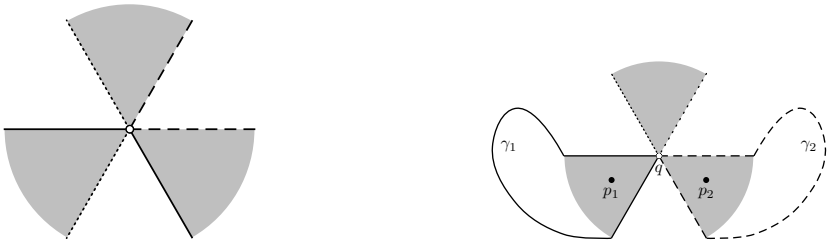
FIGURE 3. Decomposing the boundary $\partial\mathcal{Y}$ of a connected Yin set $\mathcal{Y}$ into a set of Jordan curves by disentangling the multiple boundary curves that intersect at a boundary point q (the hollow dot). This example corresponds to Figure 2(b) and (c). The shaded fan-shaped wedges represent connected components of $\mathcal{Y} \cap \mathcal{N}_r(q)$, and the white fan-shaped wedges those of $\mathcal{Y}^\perp \cap \mathcal{N}_r(q)$. Two simple curves incident to q are assigned to the same Jordan curve if they are part of the boundary of the same component of $\mathcal{Y}^\perp \cap \mathcal{N}_r(p)$.

(c) *$\mathcal{N}_r(p) \setminus \partial\mathcal{Y}$ consists of an even number of disjoint regular open sets; for two such sets sharing a common boundary, one is a subset of $\mathcal{Y}$ while the other that of $\mathcal{Y}^\perp$.*

Proof. Since $\mathcal{Y}$ is semianalytic, Definition 3.2 implies that $\mathcal{Y} \cap \mathcal{N}_r(p)$ is defined by a finite number of analytic functions $g_i : \mathbb{R}^2 \to \mathbb{R}$. By the implicit function theorem, each $g_i(\mathbf{x}) = 0$ defines a planar curve. Then (a) and (b) follow from the condition of $\mathcal{Y}$ being regular open and the condition that $r > 0$ can be as small as one wishes. Hence the local topology at a boundary point p can be characterized by the number of the aforementioned curves that intersect at p, as is shown in Figure 2.

By (a), (b), and the fact of $\mathcal{N}_r(p)$ being regular open, $\mathcal{N}_r(p) \setminus \partial\mathcal{Y}$ consists of an even number of disjoint regular open sets. Consider two such sets that share a common boundary. Suppose both of them are subsets of $\mathcal{Y}$; then it contradicts the fact of $\mathcal{Y}$ being regular. Suppose both of them are subsets of $\mathcal{Y}^\perp$; then it contradicts the fact of their common boundary being a subset of $\partial\mathcal{Y}$. Hence (c) follows. □

The above lemma on the local topology naturally yields a result on the global topology of a connected Yin set.



(a) The boundary Jordan curves have proper intersections

(b) $\mathcal{Y}$ becomes disconnected

FIGURE 4. The decomposition method shown in Figure 3 is the only valid choice to decompose the boundary of a *connected* Yin set into pairwise almost disjoint Jordan curves, because any other choice yields a contradiction.

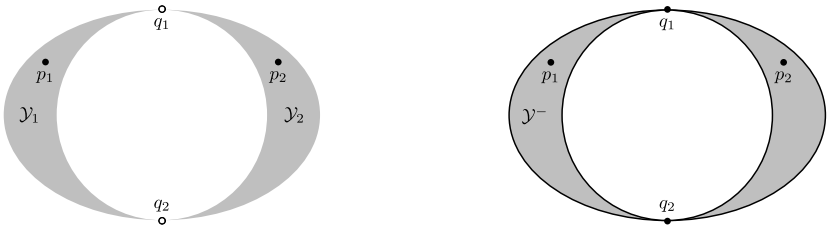
Theorem 3.9. *For a connected Yin set $\mathcal{Y} \neq \emptyset, \mathbb{R}^2$, its boundary $\partial\mathcal{Y}$ can be uniquely partitioned into a finite set of pairwise almost disjoint Jordan curves.*

Proof. Without loss of generality, we focus on a single connected component of $\partial\mathcal{Y}$. By Lemma 3.8, a boundary point $q \in \partial\mathcal{Y}$ can be classified into two types according to n_c, the number of curves that intersect at q.

If all boundary points satisfy $n_c = 1$, the condition of $\partial\mathcal{Y}$ being bounded implies that this connected component of $\partial\mathcal{Y}$ must be a single Jordan curve. Hence the statement holds trivially.

Otherwise a finite number of boundary points satisfy $n_c > 1$. Then we construct a multigraph $G_{\partial\mathcal{Y}}$ by setting its vertex set as $\{q\}$ and by obtaining its edges from dividing $\partial\mathcal{Y}$ with $\{q\}$. By Lemma 3.8(a), (b), and $\partial\mathcal{Y}$ being connected, the degree of each vertex in $G_{\partial\mathcal{Y}}$ is even. Then it follows from Theorem 2.12 that we can decompose $\partial\mathcal{Y}$ into edge-disjoint cycles. As shown in Figure 3, the decomposition is performed by requiring that, at each vertex q, two edges incident to q are assigned to the same cycle if and only if they belong to the boundary of the same connected component of $\mathcal{Y}^\perp \cap \mathcal{N}_r(q)$. Consequently, no edges in different cycles intersect properly at each self-intersection, hence the resulting Jordan curves are pairwise almost disjoint.

Finally, we show that the decomposition in Figure 3 is the only valid choice. Consider the other possibilities shown in Figure 4. If the two edges in the same cycle are not adjacent as in Figure 4(a), then two cycles would have a proper intersection at q, which contradicts the condition of the Jordan curves having no proper intersections. As for the last possibility shown in Figure 4(b), two edges in the same cycle are adjacent but they both belong to the boundary of some connected component of $\mathcal{Y} \cap \mathcal{N}_r(p)$. Then we can draw Jordan curves $\gamma_1, \gamma_2 \subset \partial\mathcal{Y}$ that contain them. The simpleness of a Jordan curve implies $\gamma_1 \neq \gamma_2$. Then there exist two points $p_1, p_2 \in \mathcal{Y}$ that belong to the bounded complements of γ_1 and γ_2, respectively. Because $\mathcal{Y}$ is connected, there exists a path within $\mathcal{Y}$ that connects p_1 and p_2. By Theorem 3.5, this path has to intersect γ_1 at some point, say p_c. The



(a) $\mathcal{Y} = \mathcal{Y}_1 \cup \mathcal{Y}_2$; $\mathcal{Y}_1 \cap \mathcal{Y}_2 = \emptyset$; $q_1, q_2 \notin \mathcal{Y}$ (b) $\mathcal{Y}^- = \mathcal{Y}_1^- \cup \mathcal{Y}_2^-$; $q_1, q_2 \in \mathcal{Y}^-$

FIGURE 5. Theorem 3.9 does not hold for the closure of a Yin set. In (a), the Yin set $\mathcal{Y}$ consists of two disjoint components $\mathcal{Y}_1$ and $\mathcal{Y}_2$ that share two common boundary points q_1 and q_2. In (b), the closure of $\mathcal{Y}$, $\mathcal{Y}^- = \mathcal{Y} \cup \partial\mathcal{Y}$, becomes connected. The solid curves indicate that $\mathcal{Y}^-$ is a regular closed set. The crucial difference is that, after the closure of $\mathcal{Y}$, the two points p_1, p_2 that previously belonged to the two disjoint Yin sets in (a) can now be joined by a path in $\mathcal{Y}^-$. Consequently, the decomposition of $\partial\mathcal{Y}^-$ into a set of pairwise almost disjoint Jordan curves is no longer unique.

construction of this path implies $p_c \in \mathcal{Y}$; the construction of γ_1 implies $p_c \in \partial\mathcal{Y}$. This is a contradiction because $\mathcal{Y}$ is open. □

The above proof hinges on the fact of a Yin set being open, so Theorem 3.9 may not hold for the closure of a Yin set. As shown in Figure 5, the decomposition of the boundary of a regular closed set is *not* unique. This is a main reason that we do not model physically meaningful regions with regular closed sets.

To relate $\mathcal{Y}$ to its boundary Jordan curves, we first define a partial order on Jordan curves.

Definition 3.10 (Inclusion of Jordan curves)**.** A Jordan curve γ_k is said to *include* γ_ℓ, written as $\gamma_k \geq \gamma_\ell$ or $\gamma_\ell \leq \gamma_k$, if and only if the bounded complement of γ_ℓ is a subset of that of γ_k. If γ_k includes γ_ℓ and $\gamma_k \neq \gamma_\ell$, we write $\gamma_k > \gamma_\ell$ or $\gamma_\ell < \gamma_k$.

Definitions 2.1 and 3.10 yield a covering relation for Jordan curves.

Definition 3.11 (Covering of Jordan curves)**.** Let $\mathcal{J}$ denote a poset of Jordan curves with inclusion as the partial order. We say γ_k *covers* γ_ℓ in $\mathcal{J}$ and write "$\gamma_k \succ \gamma_\ell$" or "$\gamma_\ell \prec \gamma_k$" if $\gamma_\ell < \gamma_k$ and no elements $\gamma \in \mathcal{J}$ satisfy $\gamma_\ell < \gamma < \gamma_k$.

Now we state the most important result of this subsection.

Theorem 3.12. *Suppose a Yin set $\mathcal{Y} \neq \emptyset, \mathbb{R}^2$ is connected. Then the Jordan curves as the unique decomposition of $\partial\mathcal{Y}$, given by the method shown in Figure* 3*, can further be uniquely oriented such that*

$$\mathcal{Y} = \bigcap_{\gamma_j \in \mathcal{J}_{\partial\mathcal{Y}}} \mathrm{int}(\gamma_j). \tag{3.3}$$

$\mathcal{J}_{\partial\mathcal{Y}}$, the set of oriented boundary Jordan curves of $\mathcal{Y}$, must be one of the two types,

$$\begin{cases} \mathcal{J}^- = \{\gamma_1^-, \gamma_2^-, \ldots, \gamma_{n_-}^-\}, & n_- \geq 1, \\ \mathcal{J}^+ = \{\gamma^+, \gamma_1^-, \gamma_2^-, \ldots, \gamma_{n_-}^-\}, & n_- \geq 0, \end{cases} \tag{3.4}$$

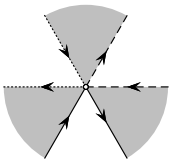


FIGURE 6. Decomposing connected $\partial\mathcal{Y}$ into a set of oriented Jordan curves. The connected Yin set $\mathcal{Y}$ is represented by shaded regions. The Jordan curves determined by the choice shown in Figure 3 can be uniquely oriented by requiring that $\mathcal{Y}$ always lies at the left of each curve.

where all γ_j^-'s are negatively oriented, mutally incomparable with respect to inclusion. For $\mathcal{J}^+$, we also have

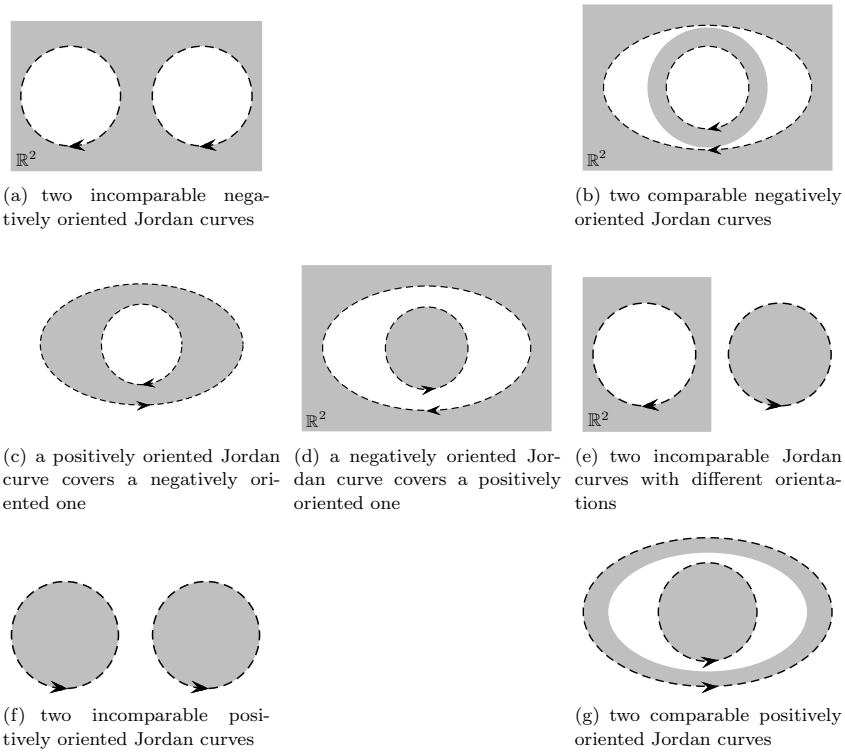
$$\forall j = 1, 2, \ldots, n_-, \quad \gamma_j^- \prec \gamma^+. \tag{3.5}$$

Proof. As in the proof of Theorem 3.9, we construct a multigraph $G_{\partial\mathcal{Y}}$ from $\partial\mathcal{Y}$. $G_{\partial\mathcal{Y}}$ is further made a directed multigraph via orienting $\partial\mathcal{Y}$ so that $\mathcal{Y}$ always lies at the left side of any oriented Jordan curve. As shown in Figure 6, the indegree and outdegree of any vertex in $G_{\partial\mathcal{Y}}$ are equal. By Theorem 2.13, there exists at least one directed cycle decomposition of $\partial\mathcal{Y}$. Then the uniqueness of the directed cycle decomposition follows from the uniqueness of orienting the Jordan curves in Figure 6.

Consider the case $\mathcal{J}_{\partial\mathcal{Y}} = \mathcal{J}^-$. For $n_- > 1$, suppose that γ_1^- and γ_2^- is comparable, as shown in Figure 7(b). According to the orientation, $\mathcal{Y}$ must lie at the left side of both γ_1^- and γ_2^-, but this is impossible because both γ_1^- and γ_2^- are part of the boundary of $\mathcal{Y}$. Consequently, (3.3) follows from Definition 3.6; see Figure 7(a).

Consider the case $\mathcal{J}_{\partial\mathcal{Y}} = \mathcal{J}^+$. If $n_- = 0$, then (3.5) holds vacuously and (3.3) holds trivially from Definition 3.6. For $n_- > 0$, suppose (3.5) did not hold for a negatively oriented Jordan curve γ_1^-. Then the almost disjointness implies that either $\gamma^+ \prec \gamma_1^-$ or they are not comparable. Suppose the former case holds. A path from one point at the left of γ^+ to another point at the left of γ_1^- must contain some points not in $\mathcal{Y}$, which contradicts the condition of $\mathcal{Y}$ being connected; see Figure 7(d). The latter case does not hold either because it contradicts the fact that γ^+ is part of the boundary of $\mathcal{Y}$; see Figure 7(e). Hence (3.5) must hold. By arguments in the previous paragraph, the negatively oriented Jordan curves must also be pairwise incomparable. Therefore, (3.3) follows from Definition 3.6; see Figure 7(c).

Suppose $\mathcal{J}_{\partial\mathcal{Y}}$ contains two positively oriented Jordan curves γ_1, γ_2. Then their almost disjointness implies that $\mathrm{int}(\gamma_1)$ is either in the unbounded complement or the bounded complement of γ_2. By similar arguments, the former contradicts the fact of $\mathcal{Y}$ being connected, as in Figure 7(f), and the latter contradicts the fact that both γ_1 and γ_2 are part of the boundary of $\mathcal{Y}$, as in Figure 7(g). Hence $\mathcal{J}_{\partial\mathcal{Y}}$ contains at most one positively oriented Jordan curve. This completes the proof. □



(a) two incomparable negatively oriented Jordan curves

(b) two comparable negatively oriented Jordan curves

(c) a positively oriented Jordan curve covers a negatively oriented one

(d) a negatively oriented Jordan curve covers a positively oriented one

(e) two incomparable Jordan curves with different orientations

(f) two incomparable positively oriented Jordan curves

(g) two comparable positively oriented Jordan curves

FIGURE 7. Enumerating all cases of two oriented almost disjoint Jordan curves γ_1 and γ_2 with respect to their orientations and inclusion relations. This is useful in proving Theorem 3.12: if γ_1 and γ_2 are in the unique decomposition of the boundary of a connected Yin set $\mathcal{Y}$ and we require that $\mathcal{Y}$ always be at the left side of both γ_1 and γ_2, then only (a) and (c) are valid, because (b), (e), and (g) contradict the fact of both γ_1 and γ_2 are part of the boundary of $\mathcal{Y}$ and (d) and (f) contradict the condition of $\mathcal{Y}$ being connected.

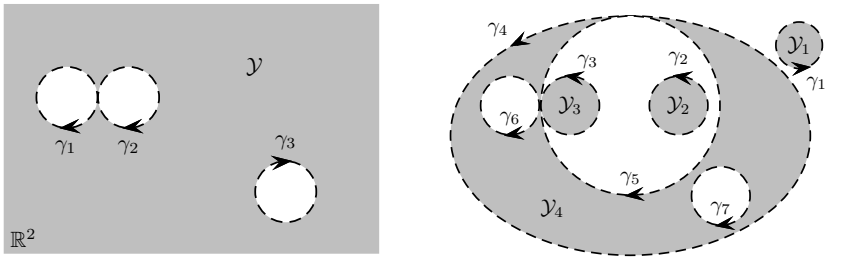
Corollary 3.13. *Each Yin set $\mathcal{Y} \neq \emptyset, \mathbb{R}^2$ can be uniquely expressed as*

$$\mathcal{Y} = \bigcup_j^{\perp\perp} \bigcap_i \operatorname{int}(\gamma_{j,i}), \tag{3.6}$$

where j is the index of connected components of $\mathcal{Y}$ and the $\gamma_{j,i}$'s are oriented Jordan curves that are pairwise almost disjoint.

Proof. The conclusion follows from applying Theorem 3.12 to each connected component of $\mathcal{Y}$. □

We illustrate (3.3) and (3.6) by the two distinct types of Yin sets in Figure 8.



(a) a connected Yin set (b) a Yin set with four connected components

FIGURE 8. Orienting boundary Jordan curves of the Yin sets in Figure 1 as illustrations of the two types of connected Yin sets classified in Theorem 3.12. In subplot (a), $\mathcal{Y} = \bigcap_{j=1}^{3} \operatorname{int}(\gamma_j)$; in subplot (b), $\mathcal{Y} = \bigcup_{i=1}^{4} \mathcal{Y}_i = \operatorname{int}(\gamma_1) \cup \operatorname{int}(\gamma_2) \cup \operatorname{int}(\gamma_3) \cup \left[\bigcap_{j=4}^{7} \operatorname{int}(\gamma_j)\right]$. By Theorem 3.12, the boundaries of the connected Yin sets $\mathcal{Y}$ in (a) and $\mathcal{Y}_4$ in (b) are of the types $\mathcal{J}^-$ and $\mathcal{J}^+$, respectively.

By results on the global topology, it is straightforward to identify the Betti numbers of a Yin set with the numbers of oriented Jordan curves in its representation.

Corollary 3.14. *The number of holes in a connected Yin set is the number of negatively oriented Jordan curves in the unique expression* (3.3)*. The number of connected components in a bounded Yin set is the number of positively oriented Jordan curves in the unique expression* (3.6)*.*

This simple result is due to the topological stratification of the Yin space and the natural correspondence of holes to negatively oriented Jordan curves.

3.5. $\mathbb{J}$**: Representing Yin sets via realizable spadjors.** Our starting point is the following acronym.

Definition 3.15. A *spadjor* is a nonempty set $\mathcal{J}_k$ of pairwise almost disjoint, piecewise-analytic oriented Jordan curves.

The following boundary-to-interior map ρ assigns to each spadjor a Yin set,

$$\rho(\mathcal{J}_k) := \bigcap\nolimits_{\gamma_i \in \mathcal{J}_k} \operatorname{int}(\gamma_i). \tag{3.7}$$

Not all spadjors are useful for representing Yin sets. The global topology of Yin sets in Theorem 3.12 and Corollary 3.13 naturally suggests that we should limit our attention to certain types of spadjors.

Definition 3.16. An *atom spadjor* is a spadjor $\mathcal{J}_k$ that consists of at most one positively oriented Jordan curve γ^+ and a finite number of negatively oriented Jordan curves γ_1^-, γ_2^-, …, $\gamma_{n_-}^-$ such that

(a) γ_j^-'s are pairwise incomparable with respect to inclusion,
(b) $\gamma_\ell^- \prec \gamma^+$ for each $\ell = 1, 2, \ldots, n_-$,
(c) $\rho(\mathcal{J}_k)$ is a connected Yin set.

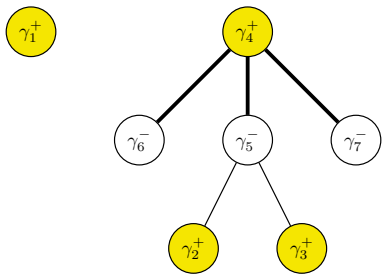


FIGURE 9. The Hasse diagram for the realizable spadjor $\mathcal{J}$ in (3.8) that represents the Yin set in Figure 8(b). The partial order is the "inclusion" relation as in Definition 3.10. A shaded circle represents a positively oriented Jordan curve while an unshaded circle a negatively oriented Jordan curve. The partition of $\mathcal{J}$ into atom spadjors consists of two steps: (a) form an atom spadjor from each shaded circle and its immediate children (if there are any) and (b) if any unshaded circles remain, group them into $\mathcal{J}^-$. These two steps correspond to (R2A-a,b) in Lemma 3.18.

Since a spadjor cannot be an empty set, $n_- = 0$ implies the presence of γ^+ and the absence of γ^+ implies $n_- > 0$. By definition, an atom spadjor is in the form of either $\mathcal{J}^-$ or $\mathcal{J}^+$ in (3.4).

Definition 3.17. A *realizable spadjor* $\mathcal{J} = \bigcup_k \mathcal{J}_k$ is the union of a finite number of atom spadjors such that $\mathcal{J}_i$ and $\mathcal{J}_j$ being distinct imply $\rho(\mathcal{J}_i) \cap \rho(\mathcal{J}_j) = \emptyset$.

Intuitively, an atom spadjor represents a connected Yin set while a realizable spadjor may represent a Yin set with multiple connected components. For example, the Yin set in Figure 8(b) can be represented by the realizable spadjor

$$\mathcal{J} = \left\{\gamma_1^+, \gamma_4^+, \gamma_6^-, \gamma_5^-, \gamma_7^-, \gamma_2^+, \gamma_3^+\right\}. \tag{3.8}$$

Lemma 3.18 concerns recovering the atom spadjors in a realizable spadjor.

Lemma 3.18. *Any realizable spajor $\mathcal{J}$ can be* uniquely *expressed as*

$$\mathcal{J} = \bigcup_{i=1}^{n} \mathcal{J}_i^+ \cup \mathcal{J}^-, \tag{3.9}$$

where the $\mathcal{J}_i^+$'s and the $\mathcal{J}^-$ are extracted from $\mathcal{J}$ as follows:

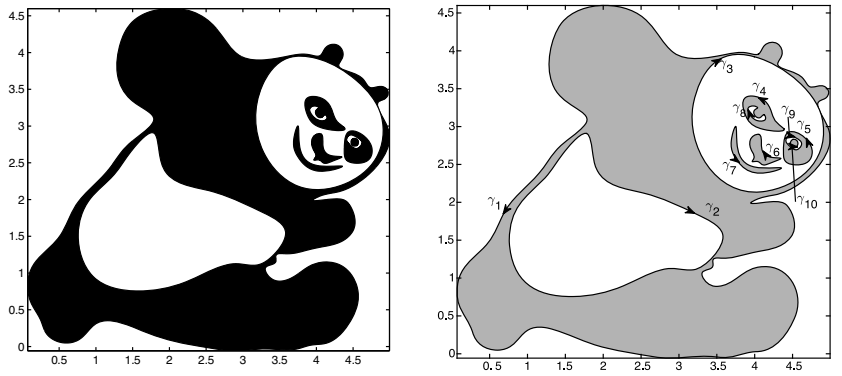
(R2A-a) *For each positively oriented Jordan curve $\gamma_i^+ \in \mathcal{J}$, form an atom spadjor $\mathcal{J}^+$ by adding γ_i^+ and all negatively oriented Jordan curves covered by γ_i^+.*

(R2A-b) *If there are negatively oriented Jordan curves left in $\mathcal{J}$, group them into an atom spadjor $\mathcal{J}^-$.*

Proof. The existence of the expression in (3.9) follows from Definition 3.17 and Theorem 3.12. As for the uniqueness, we first note that at most one atom spadjor in the form of $\mathcal{J}^-$ can be extracted from any realizable spadjor; otherwise it would contradict the condition $\rho(\mathcal{J}_i) \cap \rho(\mathcal{J}_j) = \emptyset$ in Definition 3.17. Second, $\mathcal{J}^- \neq \emptyset$ and $n > 0$ imply that the positively oriented Jordan curve γ_k^+ of any $\mathcal{J}_k^+ \subset \mathcal{J}$ is covered

by a negatively oriented element $\gamma_i^- \in \mathcal{J}^-$. Third, any negatively oriented Jordan curve not belonging to $\mathcal{J}^-$ can be *covered* by at most one $\mathcal{J}_k^+$. □

The partition of a realizable spadjor $\mathcal{J}$ into atom spadjors is best illustrated by the Hasse diagram of the poset $\mathcal{J}$ with respect to inclusion; cf. Figure 9.



(a) a panda modeled as a Yin set $\mathcal{P}$

(b) the unique representation of $\mathcal{P}$ as a realizable spadjor $\mathcal{J}$

FIGURE 10. A Yin set with complex topology and geometry. In subplot (b), the Jordan curves γ_1, γ_4, γ_5, γ_6, γ_7, and γ_{10} are positively oriented while the others are negatively oriented. The realizable spadjor is $\mathcal{J} = \bigcup_{k=1}^{6} \mathcal{J}_k$, with the atom spadjors as $\mathcal{J}_1 = \{\gamma_1, \gamma_2, \gamma_3\}$, $\mathcal{J}_2 = \{\gamma_4, \gamma_8\}$, $\mathcal{J}_3 = \{\gamma_5, \gamma_9\}$, $\mathcal{J}_4 = \{\gamma_6\}$, $\mathcal{J}_5 = \{\gamma_7\}$, and $\mathcal{J}_6 = \{\gamma_{10}\}$; all atom spadjors are of the type $\mathcal{J}^+$ in (3.4). The panda is uniquely expressed as $\mathcal{P} = \bigcup_{k=1}^{6} \mathcal{P}_k = \rho(\mathcal{J})$ with each connected component as $\mathcal{P}_k = \rho(\mathcal{J}_k)$. The picture in subplot (a) is a raster image while the curves in subplot (b) are cubic splines fit through a total of 120 points. This small amount of points demonstrates the efficiency of realizable spadjor in representing complex topology and geometry.

Definition 3.19. The *Jordan space* is the set

$$\mathbb{J} := \{\hat{0}, \hat{1}\} \cup \{\mathcal{J}\}, \tag{3.10}$$

where $\{\mathcal{J}\}$ denotes the set of all realizable spadjors and $\hat{0}$, $\hat{1}$ are two symbols satisfying

$$\rho(\hat{0}) := \emptyset, \qquad \rho(\hat{1}) := \mathbb{R}^2. \tag{3.11}$$

One can interpret $\hat{0}$ and $\hat{1}$ as atom spadjors consisting of a single oriented Jordan curve with infinitesimal diameter such that its interior goes to $\emptyset$ and $\mathbb{R}^2$, respectively.

It now makes sense to extend the definition of the boundary-to-interior map in (3.7) and (3.11) to the Jordan space.

Definition 3.20. The *boundary-to-interior map* $\rho : \mathbb{J} \to \mathbb{Y}$ associates a Yin set with each element in the Jordan space as

$$\rho(\mathcal{J}) := \begin{cases} \emptyset & \text{if } \mathcal{J} = \hat{0}; \\ \mathbb{R}^2 & \text{if } \mathcal{J} = \hat{1}; \\ \bigcup^{\perp\perp}_{\mathcal{J}_j \subset \mathcal{J}} \bigcap_{\gamma_i \in \mathcal{J}_j} \mathrm{int}(\gamma_i) & \text{otherwise,} \end{cases} \tag{3.12}$$

where the $\mathcal{J}_i$'s are atom spadjors extracted from $\mathcal{J}$ as in Lemma 3.18.

Theorem 3.21. *The boundary-to-interior map in Definition* 3.20 *is bijective.*

Proof. For the Yin sets $\emptyset$ and $\mathbb{R}^2$, (3.11) states that $\hat{0}$ and $\hat{1}$ are their preimages. For any other Yin set $\mathcal{Y} \neq \emptyset, \mathbb{R}^2$, we can uniquely decompose it as $\mathcal{Y} = \bigcup^{\perp\perp}_{\mathcal{Y}_i \subseteq \mathcal{Y}} \mathcal{Y}_i$, where the connected components $\mathcal{Y}_i$ are pairwise disjoint. By Theorem 3.12, each $\mathcal{Y}_i$ is uniquely expressed as the intersection of interiors of a number of oriented Jordan curves. Hence ρ is surjective. Also, ρ is injective because of the uniqueness in Lemma 3.18. □

Corollary 3.22. *A Yin set is uniquely represented by a realizable spadjor.*

Proof. This follows from Theorem 3.21 and Corollary 3.13. □

We sum up this section by Figure 10, where a physically meaningful region with complex topology is modeled by a fun Yin set, which is further uniquely represented by a realizable spadjor.

4. The Boolean algebra on Yin sets

After introducing the pasting map in Section 4.1, we define in Sections 4.2 and 4.3 the complementation and the meet operations on realizable spadjors to equip the Jordan space $\mathbb{J}$ as a bounded distributive lattice. Along the way, we show that these operations are counterparts to Boolean operations on the Yin space. Our theory culminates in Section 4.4. In Section 5.3, we discuss implementation issues and present a fun example of our Boolean algorithms on Yin sets.

4.1. The pasting map of realizable spadjors.

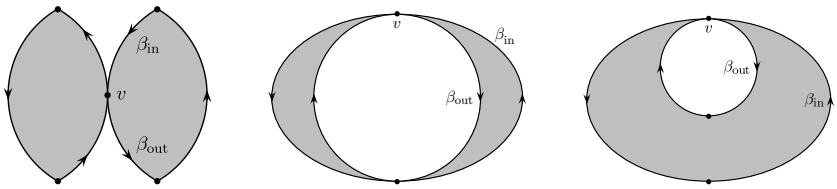
The following cutting map is trivial, but its inverse in Lemma 4.2 is not. They are crucial for the complementation and the meet operations in Sections 4.2 and 4.3.

Definition 4.1. For a realizable spadjor $\mathcal{J}$, let V denote a finite point set that contains all intersections of the Jordan curves in $\mathcal{J}$. The associated *cutting map* or *segmentation map* S_V assigns to $\mathcal{J}$ a set E of oriented paths obtained by dividing the oriented Jordan curve in $\mathcal{J}$ at points in V. The set of (oriented) paths $E = S_V(\mathcal{J})$ is called a *segmented realizable spadjor*.

Lemma 4.2. *The realizable spadjor $\mathcal{J} = S_V^{-1}(E)$ corresponding to a segmented realizable spadjor E can be uniquely constructed as follows:*

(S2R-a) *Remove all self-loops in E and insert them into $\mathcal{J}$.*

(S2R-b) *Start with a path $\beta_{in} \in E$ and denote by $v \in V$ its ending point. If there exists only one path whose starting point is v, call it β_{out} and append it to β_{in}. Otherwise, set β_{out} to be the edge that starts at v and of which the positive counterclockwise angle $\angle\beta_{out}v\beta_{in}$ is the smallest. Repeat the above conditional to grow the path until it becomes a loop γ_1, and remove from E all paths that constitute γ_1.*



(a) Two components and one improper intersection

(b) Two components and two improper intersections

(c) One component and one improper intersection

FIGURE 11. Illustrating key steps (S2R-b,c) of the pasting map in Lemma 4.2. The shaded region represents a Yin set whose boundary consists of two Jordan curves with an improper intersection at v. In each subplot, the solid dots represent points of V while the directed paths constitute E. Starting from β_{in}, we pick β_{out} to grow the starting path because $\angle\beta_{\text{out}}v\beta_{\text{in}}$ is the smallest counterclockwise angle among those of the two out-edges; this condition in (S2R-b) is different from that in Figure 3. In subplot (c), the loop γ_1 resulting from (S2R-b) consumes all paths. Hence in (S2R-c) we divide it into two Jordan loops to fulfill the representation invariant of realizable spadjors.

(S2R-c) *If γ_1 is a Jordan curve, add it into $\mathcal{J}$; otherwise divide γ_1 into Jordan curves and/or self-loops and add them into $\mathcal{J}$.*

(S2R-d) *Repeat* (S2R-b,c) *to add other Jordan curves into $\mathcal{J}$ until E becomes empty.*

Proof. First we note that Theorem 3.9 is not applicable here because the Yin set $\rho(\mathcal{J})$ may be disconnected. By Definition 3.6, the Yin set $\rho(\mathcal{J})$ always lies at the left of any oriented path $\beta \in E$; the steps (S2R-a,b) are sufficient to fulfill this invariant. The choice for growing the path in (S2R-b) also avoids potential proper intersections of Jordan curves in $\mathcal{J}$; see Figure 4(a). However, as suggested by Figure 4(b) and the proof of Theorem 3.9, the loop γ_1 might not be a single Jordan curve, hence we need to divide it into Jordan curves and/or self-loops in (S2R-c); see Figure 11. The uniqueness of the constructed realizable spadjor follows from the uniqueness in Corollary 3.22. □

The procedures in Lemma 4.2 define the inverse of the cutting map, which we refer to as the *pasting map of realizable spadjors*.

4.2. **Complementation: A unitary operation on $\mathbb{J}$.** The following is an easy result on the local topology of regular sets.

Corollary 4.3. *Suppose $\mathcal{Y}$ is a regular open or regular closed set. Then a point $p \in \mathbb{R}^2$ is a boundary point of $\mathcal{Y}$ if and only if, for any sufficiently small $r > 0$, the open ball $\mathcal{N}_r(p)$ centered at p with radius r contains both points in $\mathcal{Y}$ and $\mathcal{Y}^\perp$.*

Proof. The necessity follows directly from Lemma 3.8(c); we only prove the sufficiency. If r is sufficiently small, there are only seven cases for the type of points contained in $\mathcal{N}_r(p)$: (i) $\mathcal{Y}$, (ii) $\mathcal{Y}^\perp$, (iii) $\partial\mathcal{Y}$, (iv) $\mathcal{Y}$ and $\mathcal{Y}^\perp$, (v) $\mathcal{Y}$ and $\partial\mathcal{Y}$, (vi) $\mathcal{Y}^\perp$

and $\partial\mathcal{Y}$, and (vii) all three sets. Because of the regularity, cases (iii)–(vi) are impossible. By definitions in Section 3.1, (i) implies an interior point and (ii) implies an exterior point. Hence (vii) must imply a boundary point. In other words, the presence of $\mathcal{Y}$ and $\mathcal{Y}^\perp$ in $\mathcal{N}_r(p)$ dictates that of $\partial\mathcal{Y}$ in $\mathcal{N}_r(p)$. □

Corollary 4.3 and Lemma 4.2 motivate our complementation operation on $\mathbb{J}$.

Definition 4.4. The *complementation operation* $' : \mathbb{J} \to \mathbb{J}$ is defined as

$$\mathcal{J}' := \begin{cases} \hat{1} & \text{if } \mathcal{J} = \hat{0}; \\ \hat{0} & \text{if } \mathcal{J} = \hat{1}; \\ \left(S_V^{-1} \circ R \circ S_V\right) \mathcal{J} & \text{otherwise,} \end{cases} \tag{4.1}$$

where V is the set of improper intersections of Jordan curves in $\mathcal{J}$ and where the orientation-reversing map R reverses the orientation of each path in the segmented realizable spadjor $S_V(\mathcal{J})$.

Lemma 4.5. *The complementation operation in Definition* 4.4 *satisfies*

$$\forall \mathcal{J} \in \mathbb{J}, \qquad \rho(\mathcal{J}') = (\rho(\mathcal{J}))^\perp. \tag{4.2}$$

Proof. By definition, a regular set $\mathcal{Y}$ satisfies $(\mathcal{Y}^\perp)^\perp = \mathcal{Y}$, which, together with Corollary 4.3, imply that $\mathcal{Y}$ and $\mathcal{Y}^\perp$ have exactly the same boundary. By Corollary 3.13 and Definition 3.6, the realizable spadjors representing $\mathcal{Y}$ and $\mathcal{Y}^\perp$ are the same set of Jordan curves except that each pair of corresponding Jordan curves has different orientations, which justifies the necessity of the orientation-reversing map R. More precisely, the conjugate of R by the cutting map S_V is needed here because one or multiple improper intersections of two Jordan curves may dictate that the paths constituting oriented Jordan curves in $\mathcal{J}$ be reorganized in order to properly represent $\mathcal{Y}^\perp$. For example, the calculation of $\mathcal{Y}^\perp$ for the Yin set $\mathcal{Y}$ in Figure 11(b) involves not only reversing the orientation of each path, but also reorganizing these paths into different Jordan curves. □

4.3. **Meet: A binary operation on $\mathbb{J}$.** To define the meet of two realizable spadjors $\mathcal{J}$ and $\mathcal{K}$, we cut them by S_V, select those paths that are on the boundary of the Yin set $\rho(\mathcal{J}) \cap \rho(\mathcal{K})$, and paste the set of selected paths by S_V^{-1} to form the result. Lemma 4.6 and Corollary 4.7 tell us which paths we should choose.

Lemma 4.6. *Denote $\mathcal{Y} := \operatorname{int}(\sigma_1) \cap \operatorname{int}(\sigma_2)$, where σ_1 and σ_2 are two oriented Jordan curves. For a curve β satisfying $\beta \subseteq \sigma_1$ and $\beta \subset \mathbb{R}^2 \setminus \sigma_2$, we have $\beta \subseteq \partial\mathcal{Y}$ if and only if $\beta \subset \operatorname{int}(\sigma_2)$. For a curve β satisfying $\beta \subseteq \sigma_1 \cap \sigma_2$, we have $\beta \subseteq \partial\mathcal{Y}$ if and only if the direction of β induced from the orientation of σ_1 is the same as that induced from the orientation of σ_2.*

Proof. We only prove the first statement since the second one can be similarly shown. By Definition 3.6, $\operatorname{int}(\sigma_1)$ and $\operatorname{int}(\sigma_2)$ are both Yin sets. It follows from Theorem 3.4 that $\mathcal{Y}$ is also a Yin set. By Corollary 4.3, it suffices to show that a small open ball $\mathcal{N}_r(p)$ centered at $p \in \beta$ contains both points in $\mathcal{Y}$ and $\mathcal{Y}^\perp$ if and only if $\beta \subset \operatorname{int}(\sigma_2)$. As shown in Figure 12, $p \in \beta$ and $\beta \subseteq \sigma_1$ imply that $\mathcal{N}_r(p) \cap \operatorname{int}(\sigma_1) \neq \emptyset$; then the condition $\beta \subset \operatorname{int}(\sigma_2)$ implies $\mathcal{N}_r(p) \cap \mathcal{Y} \neq \emptyset$. DeMorgan's law (2.6) yields

$$\mathcal{Y}^\perp = [\operatorname{int}(\sigma_1) \cap \operatorname{int}(\sigma_2)]^\perp = \operatorname{int}(\sigma_1)^\perp \cup^{\perp\perp} \operatorname{int}(\sigma_2)^\perp \neq \emptyset,$$

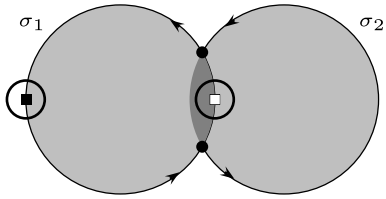


FIGURE 12. In proving Lemma 4.6, we consider $\mathcal{Y} := \mathrm{int}(\sigma_1) \cap \mathrm{int}(\sigma_2)$, where σ_1 and σ_2 are two oriented Jordan curves. By Corollary 4.3, $\beta \subset \partial\mathcal{Y}$ if and only if for any point $p \in \beta$ (the open square), any sufficiently small neighborhood of p contains both points in $\mathcal{Y}$ and $\mathcal{Y}^\perp$. On the other hand, if a point q (the filled square) is not in σ_2, then it is definitely not on the boundary of $\mathcal{Y}$.

which, together with $\beta \subseteq \sigma_1$, imply $\mathcal{N}_r(p) \cap \mathcal{Y}^\perp \neq \emptyset$. By Corollary 4.3, we have $\beta \subset \partial\mathcal{Y}$. Conversely, $\beta \not\subset \mathrm{int}(\sigma_2)$, $\beta \subset \mathbb{R}^2 \setminus \sigma_2$, and $\beta \subseteq \sigma_1$ imply that $\mathcal{N}_r(q) \cap \mathcal{Y} = \emptyset$ for all $q \in \beta$. These arguments are illustrated in Figure 12. □

Hereafter we write the union of all Jordan curves in a realizable spadjor as

$$P(\mathcal{J}) := \bigcup_{\gamma_i \in \mathcal{J}} \gamma_i, \tag{4.3}$$

which is clearly a subset of $\mathbb{R}^2$.

Corollary 4.7. *Denote $\mathcal{Y} := \rho(\mathcal{J}) \cap \rho(\mathcal{K})$, where $\mathcal{J}$ and $\mathcal{K}$ are two realizable spadjors. For a curve β satisfying $\beta \subseteq P(\mathcal{J})$ and $\beta \subset \mathbb{R}^2 \setminus P(\mathcal{K})$, we have $\beta \subseteq \partial\mathcal{Y}$ if and only if $\beta \subset \rho(\mathcal{K})$. For a curve β satisfying $\beta \subseteq P(\mathcal{J}) \cap P(\mathcal{K})$, we have $\beta \subseteq \partial\mathcal{Y}$ if and only if the direction of β induced from the orientation of $\mathcal{J}$ is the same as that induced from the orientation of $\mathcal{K}$.*

Proof. This follows directly from Lemma 4.6. □

Definition 4.8. The *meet of two realizable spadjors* $\mathcal{J}$ and $\mathcal{K}$ is a binary operation $\wedge : \mathbb{J} \times \mathbb{J} \to \mathbb{J}$ defined as

$$\mathcal{J} \wedge \mathcal{K} = \begin{cases} \hat{0} & \text{if } \mathcal{K} = \hat{0}; \\ \mathcal{J} & \text{if } \mathcal{K} = \hat{1}; \\ S_V^{-1}(E) & \text{otherwise,} \end{cases} \tag{4.4}$$

where the pasting map S_V^{-1} is defined in Lemma 4.2, and the directed multigraph (V, E) is constructed as follows:

(MRS-a) The set $\mathcal{I} := P(\mathcal{J}) \cap P(\mathcal{K})$ may contain paths and isolated points. Initialize V as an empty set; add into V all path endpoints and isolated points in $\mathcal{I}$.

(MRS-b) Cut $\mathcal{J}$ with points in V, and we have a set of paths $\{\beta_i\} = S_V(\mathcal{J})$. Initialize E as an empty set.

(MRS-c) For each β_i, add it to E if β_i minus its endpoints is contained in $\rho(\mathcal{K})$, or if there exists $\beta_j \subset P(\mathcal{K})$ such that $\beta_j = \beta_i$ and they have the same direction. In particular, if β_i is a Jordan curve that satisfies either of the above conditions, we insert β_i as a self-loop into E.

(MRS-d) For each $\beta_j \subset S_V(\mathcal{K})$, add it to E if β_j minus its endpoints is contained in $\rho(\mathcal{J})$.

Lemma 4.9. *The meet operation in Definition* 4.8 *satisfies*

$$\forall \mathcal{J}, \mathcal{K} \in \mathbb{J}, \qquad \rho(\mathcal{J} \wedge \mathcal{K}) = \rho(\mathcal{J}) \cap \rho(\mathcal{K}). \tag{4.5}$$

Proof. Denote $\mathcal{Y}_\cap := \rho(\mathcal{J}) \cap \rho(\mathcal{K})$. It follows from Corollary 4.7 that each path $\beta_i \subset S_V(\mathcal{J})$ is added to E in step (MRS-c) if and only if $\beta_i \subset \partial\mathcal{Y}_\cap$; similarly, each curve $\beta_j \subset S_V(\mathcal{K})$ is added to E in step (MRS-d) if and only if $\beta_j \subset \partial\mathcal{Y}_\cap$. Hence the union of the vertices and edges in G constitute the boundary of $\mathcal{Y}_\cap$. Furthermore, by the difference between (MRS-c) and (MRS-d), each edge on $\partial\mathcal{Y}_\cap$ is inserted into E only once. Therefore, E contains and only contains points on $\partial\mathcal{Y}_\cap$. The proof is then completed by Lemma 4.2 and Corollary 3.22. □

4.4. **The Yin space $\mathbb{Y}$ and the Jordan space $\mathbb{J}$ are isomorphic.** The join operation can be expressed by the meet operation and the complementation operation; this is also true for all other Boolean operations.

Definition 4.10. The *join of two realizable spadjors* $\mathcal{J}$ and $\mathcal{K}$ is a binary operation $\vee : \mathbb{J} \times \mathbb{J} \to \mathbb{J}$ defined as

$$\forall \mathcal{J}, \mathcal{K} \in \mathbb{J}, \qquad \mathcal{J} \vee \mathcal{K} := (\mathcal{J}' \wedge \mathcal{K}')'. \tag{4.6}$$

The following theorem is the theoretical culmination of this paper.

Theorem 4.11. *The Boolean algebras* $(\mathbb{J}, \vee, \wedge, ', \hat{0}, \hat{1})$ *and* $(\mathbb{Y}, \cup^{\perp\perp}, \cap, ^\perp, \emptyset, \mathbb{R}^2)$ *are isomorphic under the boundary-to-interior map* ρ *in Definition* 3.20.

Proof. This follows directly from Lemmas 4.5 and 4.9, Definitions 3.20 and 4.10, and DeMorgan's law (2.6). □

As the desired consequence, we have reduced the two-dimensional problems $\cup^{\perp\perp}$, $\cap$, and $^\perp$ to the one-dimensional problems $\vee$, $\wedge$, and $'$.

5. Boolean algorithms on Yin sets

Purely algebraic and constructive as they are, Definitions 4.4, 4.8, and 4.10 already constitute a complete set of Boolean algorithms on Yin sets. In this section we discuss important details on algorithmic design and implementations, state main results on the analysis of algorithmic complexity, and perform a variety of test cases to validate our theory and to verify our implementation. In particular, the utilities of our Boolean algorithms in tracking deforming fluid phases are demonstrated by results of a standard benchmark test.

We do not expect the reader to be able to implement our Boolean algorithms after reading this section. Instead, the exposition in this section is only intended to give the reader a relatively complete picture on *theoretical* aspects of Boolean algorithms on the Yin space. We will fully explain the algorithmic details of our implementation in another separate paper.

5.1. **User-friendly design.** The input is an indispensable part of an algorithm, and a user-friendly design of input parameters adds to the appealingness of an algorithm.

In our implementation, the data structure that represents Yin sets is designed to be a straightforward orchestration of the realizable spadjor, i.e., a set of point

arrays where each array represents a polygon and the direction of points in the array indicates the orientation of the polygon. To input Yin sets, the user does not need to specify the inclusion relations of the boundary Jordan curves; we have encapsulated the computation of this information inside the algorithms. So long as each input is indeed a realizable spadjor, the algorithm returns the correct result. In the meantime, the user can choose to output the results in the same format as the input; she can also output the Hasse diagram of the results with respect to the inclusion relation. These designs make the software interface simple, flexible, and user-friendly.

5.2. **Robustness.** Our algorithm is distinguished from other Boolean algorithms in that the user is given explicit control on the uncertainty of Boolean operations.

Given a small positive real number ϵ, we define two points to be the same point if their distance is smaller than ϵ. A consistent enforcement of this definition and its implications across the entire package makes our implementation robust and provides an effective mechanism to handle various degenerate cases that characterize topological changes.

It is well known in the community of computational geometry that intersecting line segments might lead to unavoidable self-inconsistencies and cause a program to abort at the run time [16]. The mathematical core of this difficulty is the potentially arbitrary ill-conditioning of intersecting line segments in an unlimited range of length scales. Fortunately, in the context of numerically simulating multiphase flows, there always exists a length scale below which finer details are not needed. Hence this uncertain parameter ϵ is not only a device of flexibility and convenience, but more importantly a simple solution of the aforementioned robustness problem in computational geometry.

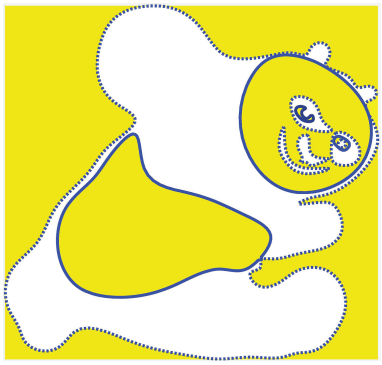
5.3. **Implementation.** Implementing Boolean operations in Section 4 reduce to two well-studied problems in computational geometry: one is to determine the relative positions of a set of points to a simple polygon, and the other is to compute intersections of the edges of two simple polygons.

We solve these two problems based on the plane sweep idea in [2], [7, Chap. 2], but our solutions differ from current methods in several aspects. First, we enforce the uncertainty criterion in Section 5.2 to deal with the overlapping of line segments for robust floating-point calculations. Second, we modify traditional algorithms to cater to our problems. For example, the point-in-polygon algorithm in [24, Sec. 7.4] designed to determine the relative position of a *single* point to a polygon with n edges has complexity $O(n)$. Repeatedly applying this algorithm to q points would have the overall complexity $O(qn)$; this is not optimal if the number of points is large. Based on the idea of a plane sweep, we developed another algorithm with complexity $O((n+q)\log n)$.

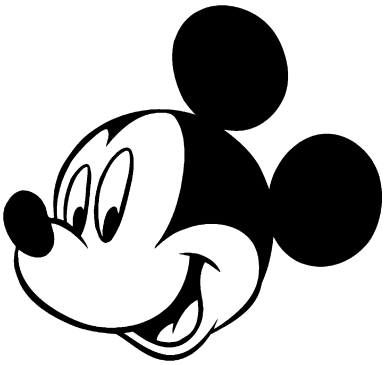
We have implemented these algorithms and tested them thoroughly by a series of test cases; some fun examples are shown in Figure 13.

5.4. **Complexity.** In discussing complexity, we restrict realizable spadjors to be constituted only by *linear* polygons.

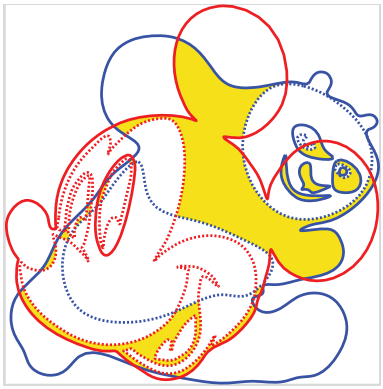
Let n denote the total number of points in two realizable spadjors that represent two Yin sets. Let I denote the number of their intersections. It can be shown that the complexity of our implementation of the meet operation in Definition 4.8 is



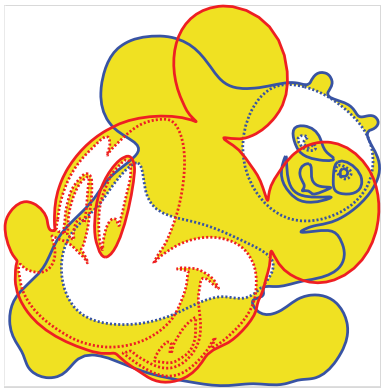
(a) $\mathcal{P}^{\perp}$: the exterior of the panda $\mathcal{P}$ in Figure 10 obtained by the complementation operation in Definition 4.4.



(b) a Mickey mouse modeled as a Yin set $\mathcal{M}$



(c) $\mathcal{M} \cap \mathcal{P}$: intersection of $\mathcal{M}$ and $\mathcal{P}$ obtained by the meet operation in Definition 4.8.

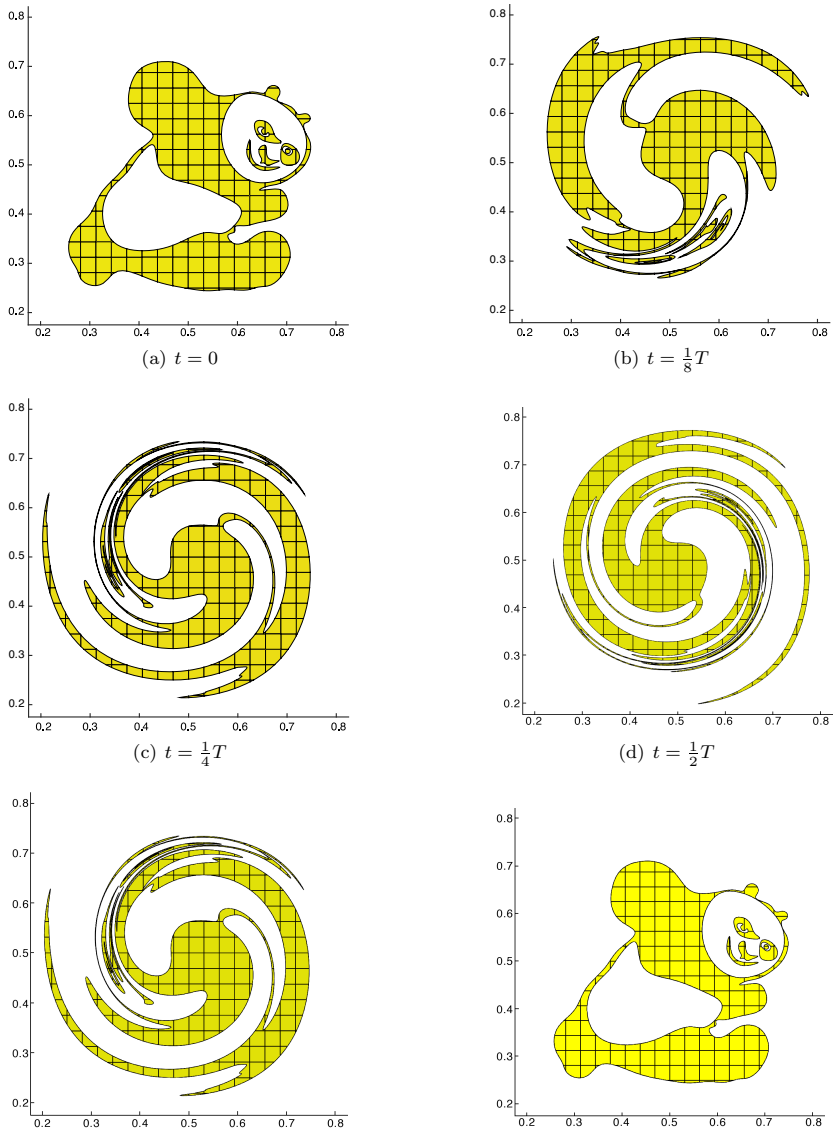


(d) $\mathcal{M} \cup^{\perp\perp} \mathcal{P}$: regularized union of $\mathcal{M}$ and $\mathcal{P}$ obtained by the join operation in Definition 4.10.

FIGURE 13. Results of testing Boolean algorithms on Yin sets with complex topology and geometry. In subplots (a), (c), and (d), a solid line represents a positively oriented Jordan curve, a dotted line a negatively oriented Jordan curve, and a shaded region the result of a Boolean operation. There are several improper intersections of the Jordan curves in the realizable spadjor that represents the panda.

$O((n + I) \log n)$. As for the complementation operation, the complexity is $O(n)$, where now n denotes the number of points in the single spadjor.

Our algorithm of constructing the Hasse diagram for a realizable spadjor has the worst-case complexity $O(m(m+l) \log l)$, where m is the number of polygons in the spadjor and l is the maximum number of points of these polygons.



(a) $t = 0$ (b) $t = \frac{1}{8}T$

(c) $t = \frac{1}{4}T$ (d) $t = \frac{1}{2}T$

(e) $t = \frac{3}{4}T$ (f) $t = T$

FIGURE 14. Results of the cubic MARS method for the vortex shear test [48, §5.2] with $T = 8$, $h = \frac{1}{32}$, $h_L = 0.2h$, $r_{\mathsf{tiny}} = 0.01$, and the panda in Figure 10(a) as the initial condition. Over all time steps, the number of connected components remains a constant and the number of holes in each connected component also remains a constant. These Betti numbers are returned in constant time.

5.5. **Applications to interface tracking.** As discussed in Section 1.2, a deforming fluid phase is usually represented by a point set $\mathcal{M}(t) := \{\mathbf{x} : f(\mathbf{x}, t) = 1\}$, where f is the color function as defined in (1.2). Then the interface tracking (IT) problem can be regarded as the determination of $\mathcal{M}(T)$ from the initial condition $\mathcal{M}(t_0)$ and the given velocity field $\mathbf{u}(\mathbf{x}, t)$, $t \in [t_0, T]$. This approach is adopted for the error analysis of IT methods under the Lebesgue measure [49]. However, to deal with topological issues, it is necessary to refine the modeling of fluid phases via Yin sets so that the IT problem can be more precisely defined.

Definition 5.1 (the IT problem)**.** The *interface tracking problem* is the determination of the final loci of the fluid phase $\mathcal{M}(T) \in \mathbb{Y}$ by passively advecting $\mathcal{M}(t_0) \in \mathbb{Y}$, the initial loci of the phase, with a velocity field $\mathbf{u} : \mathbb{R}^{\mathsf{D}} \times [t_0, T] \to \mathbb{R}^{\mathsf{D}}$, where $\mathbf{u}$ is continuous in time $t \in [t_0, T]$ and piecewise Lipschitz continuous in space.

From this viewpoint, fluid phases deform according to the action of the flow map of the velocity field upon the Yin space. Here we only require piecewise Lipschitz continuity on the velocity field so that fluid modeling via Yin sets admits topological changes such as merging. On the other hand, when the flow map is a homeomorphism, all topological information should be preserved by the flow map.

We have coupled the recent cubic MARS method [48] to the Boolean algorithms on Yin sets; the result is a more versatile method capable of handling complex topology on very coarse grids. Given $\mathcal{M}^n \approx \mathcal{M}(t_n)$, major steps of our IT method are as follows:

(IT-1) Apply the flow map to vertices of linear polygons that represent $\mathcal{M}^n \in \mathbb{Y}$ to obtain their images.
(IT-2) Add or delete points between two adjacent markers if the distance of their images is too large or too small, respectively, so that all distances of adjacent marker images are within the interval $[r_{\text{tiny}} h_L, h_L]$.
(IT-3) Connect the marker images to obtain a Yin set $\mathcal{M}^{n+1}$ as the *global* solution.
(IT-4) Calculate the Betti numbers of $\mathcal{M}^{n+1}$ if needed.
(IT-5) Intersect $\mathcal{M}^{n+1}$ to fixed control volumes of length h to obtain *local* solutions.

In step (IT-4), the Betti numbers of the tracked fluid phase are calculated in $O(1)$ time. To the best of our knowledge, no IT methods are even capable to calculate these Betti numbers. We also emphasize that the local solutions are important in coupling IT with high-order numerical solvers of the fluid phase. The reader is referred to [48] for other details.

In Table 1 and Figure 14, we show results of the standard vortex-shear test to demonstrate that (1) the linear representation of the panda yields second-order convergence rates IT, and (2) the complex topology is preserved very well by the homeomorphic flow map, i.e., the Betti numbers of the fluid phase remain constants during the whole simulation. The details of this coupled method will be reported in another paper.

6. Conclusion

We have introduced the problem of fluid modeling in multiphase flows as a counterpart of solid modeling in CAD, and have proposed to solve this problem via the Yin space, a topological space with a simple, efficient, and complete Boolean algebra. Under this framework, topological changes of Yin sets can be characterized

TABLE 1. Geometric errors and convergence rates of the vortex-shear test [48, §5.2]. The parameters of the cubic MARS method are set to $h_L = 0.2h$ and $r_{\text{tiny}} = 0.01$. For any given h, we set the initial condition to a Yin set with piecewise linear boundaries that approximates the panda, compute volume fractions of the panda inside the control volumes, run the vortex-shear test to the ending time, calculate the difference of the two sets of volume fractions at the initial time and the ending time, and sum up the difference to obtain the geometric error. Over all time steps, the zeroth Betti number of the yellow region is 6, the first Betti number of the largest connected component is 2, those of components homeomorphic to the two eyes is 1, and those of all other connected components are 0.

$h = \frac{1}{32}$	rate	$h = \frac{1}{64}$	rate	$h = \frac{1}{128}$	rate	$h = \frac{1}{256}$
2.11e-03	2.61	3.46e-04	2.70	5.33e-05	2.88	7.23e-06

and handled naturally, and topological information such as Betti numbers can be extracted in constant time.

Several prospects for future research follow. The theory and algorithms on the Yin space can be generalized to 2-manifolds in a straightforward manner. Another generalization of Yin sets to three dimensions is currently a work in progress. Finally, it would be exciting to couple Yin sets with high-order finite-volume methods [46] to form fourth- and higher-order solvers for simulating multiphase flows such as free-surface flows and fluid-structure interactions.

ACKNOWLEDGMENTS

The authors thank Difei Hu for digitizing the panda image, and Sen Li for generating the results of the vortex-shear test.

REFERENCES

[1] Y. Bazilevs, K. Takizawa, and T. E. Tezduyar, *Computational Fluid-structure Interaction: Methods and Applications*, Wiley Series in Computational Mechanics, Wiley, 2013.
[2] J. L. Bentley and T. A. Ottmann, *Algorithms for reporting and counting geometric intersections*, IEEE Trans. Comput. **C-28** (1979), 643–647.
[3] H. Bieri, *Nef polyhedra: A brief introduction*, Geometric modelling, 1995, pp. 43–60.
[4] B. Bollobás, *Modern graph theory*, corrected, Graduate Texts in Mathematics, vol. 184, Springer-Verlag, New York, 2008. ISBN:0-387-98488-7.
[5] S. Burris and H. P. Sankappanavar, *A course in universal algebra*, The Millennium, Springer, 2012. ISBN:978-0-9880552-0-9.
[6] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs, *Isogeometric Analysis: Toward Integration of CAD and FEA*, John Wiley & Sons, Ltd., Chichester, 2009. MR3618875
[7] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, 3rd ed., Springer-Verlag, Berlin, 2008. MR2723879
[8] S. Givant and P. Halmos, *Introduction to Boolean Algebras*, Undergraduate Texts in Mathematics, Springer, New York, 2009. MR2466574
[9] G. Grätzer, *Lattice Theory: First Concepts and Distributive Lattices*, Dover, 2009.
[10] G. Greiner and K. Hormann, *Efficient clipping of arbitrary polygons*, ACM Transactions on Graphics **17** (1998), no. 2, 71–83.

[11] P. Hachenberger, L. Kettner, and K. Mehlhorn, *Boolean operations on 3D selective Nef complexes: data structure, algorithms, optimized implementation and experiments*, Comput. Geom. **38** (2007), no. 1-2, 64–99, DOI 10.1016/j.comgeo.2006.11.009. MR2329542

[12] C. W. Hirt and B. D. Nichols, *Volume of fluid (VOF) method for the dynamics of free boundaries*, J. Comput. Phys. **39** (1981), 201–225.

[13] E. V. Huntington, *Sets of independent postulates for the algebra of logic*, Trans. Amer. Math. Soc. **5** (1904), no. 3, 288–309, DOI 10.2307/1986459. MR1500675

[14] C. Jordan, *Cours D'Aanlyse l'École Polytechnique*, Paris, 1887, pp. 587–594.

[15] T. Kaczynski, K. Mischaikow, and M. Mrozek, *Computational Homology*, Applied Mathematical Sciences, vol. 157, Springer-Verlag, New York, 2004. MR2028588

[16] L. Kettner, K. Mehlhorn, S. Pion, S. Schirra, and C. Yap, *Classroom examples of robustness problems in geometric computations*, Comput. Geom. **40** (2008), no. 1, 61–78, DOI 10.1016/j.comgeo.2007.06.003. MR2392653

[17] K. Kuratowski and A. Mostowski, *Set theory*, Second, completely revised edition, North-Holland Publishing Co., Amsterdam-New York-Oxford; PWN—Polish Scientific Publishers, Warsaw, 1976. With an introduction to descriptive set theory; Translated from the 1966 Polish original; Studies in Logic and the Foundations of Mathematics, Vol. 86. MR0485384

[18] Y. D. Liang and B. A. Barsky, *An analysis and algorithm for polygon clipping*, Comm. ACM **26** (1983), no. 11, 868–877, DOI 10.1145/182.358439. MR784119

[19] Y. K. Liu, X. Q. Wang, S. Z. Bao, M. Gombosi, and B. Zalik, *An algorithm for polygon clipping, and for determining polygon intersections and unions*, Computers & Geosciences **33** (2007), 589–598.

[20] H. M. MacNeille, *Partially ordered sets*, Trans. Amer. Math. Soc. **42** (1937), no. 3, 416–460, DOI 10.2307/1989739. MR1501929

[21] F. Martinez, C. Ogayar, J. R. Jimenez, and A. J. Rueda, *A simple algorithm for Boolean operations on poygons*, Advances in Engineering Software **64** (2013), 11–19.

[22] J. R. Munkres, *Elements of Algebraic Topology*, Addison-Wesley Publishing Company, Menlo Park, CA, 1984. MR755006

[23] W. Nef, *Beiträge zur Theorie der Polyeder* (German), Mit Anwendungen in der Computergraphik; Beiträge zur Mathematik, Informatik und Nachrichtentechnik, Band 1, Herbert Lang, Bern, 1978. MR0500548

[24] J. O'Rourke, *Computational geometry in c*, Second, Cambridge University Press, New York, NY, USA, 1998.

[25] S. Osher and J. A. Sethian, *Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations*, J. Comput. Phys. **79** (1988), no. 1, 12–49, DOI 10.1016/0021-9991(88)90002-2. MR965860

[26] Y. Peng, J.-H. Yong, W.-M. Dong, H. Zhang, and J.-G. Sun, *A new algorithm for Boolean operations on general polygons*, Computers & Graphics **29** (2005), 57–70.

[27] J. Qian and C. K. Law, *Regimes of coalescence and separation in droplet collision*, J. Fluid Mech. **331** (1997), 59–80.

[28] A. A. G. Requicha, *Mathematical models of rigid solid objects*, Technical Report 28, The University of Rochester, Rochester NY, 1977.

[29] A. A. G. Requicha and H. B. Voelcker, *Solid modeling: current status and research directions*, IEEE Computer Graphics and Applications **3** (1983), no. 7, 25–37.

[30] A. G. Requicha, *Representations for rigid solids: Theory, methods, and systems*, ACM Computing Surveys **12** (1980), 437–464. `http://doi.acm.org/10.1145/356827.356833`.

[31] A. G. Requicha and R. B. Tilove, *Mathematical Foundations of Constructive Solid Geometry: General Topology of Closed Regular Sets*, University of Rochester, Rochester, N.Y., 1978. `http://hdl.handle.net/1802/1209`.

[32] M. Rivero and F. R. Feito, *Boolean operations on general planar polygons*, Computers & graphics **24** (2000), 881–896.

[33] H. Sagan, *An elementary proof that Schoenberg's space-filling curve is nowhere differentiable*, Math. Mag. **65** (1992), no. 2, 125–128, DOI 10.2307/2690494. MR1160715

[34] P. Saveliev, *Topology Illustrated*, Peter Saveliev, 2016. ISBN: 978-1495188756.

[35] V. Shapiro, *Solid Modeling*, Handbook of computer aided geometric design, North-Holland, Amsterdam, 2002, pp. 473–518, DOI 10.1016/B978-044451104-1/50021-6. MR1928553

[36] L. J. Simonson, *Industrial strength polygon clipping: A novel algorithm with applications in VLSI CAD*, Computer-Aided Design **42** (2010), no. 12, 1189–1196.

[37] R. P. Stanley, *Enumerative Combinatorics. Volume 1*, 2nd ed., Cambridge Studies in Advanced Mathematics, vol. 49, Cambridge University Press, Cambridge, 2012. MR2868112
[38] E. E. Sutherland and G. W. Hodgeman, *Rentrant polygon clipping*, Communications of the ACM **17** (1974), no. 1, 32–42.
[39] A. Tarski, *Über additive and multiplikative Mengenkörper und Mengenfunktionen*, Sprawozdania z Posiedzeń Towarzystwa Naukowego Warszawskiego, Wydzial III Nauk Matematyczno-fizycznych **30** (1937), 151–181.
[40] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, and Y.-J. Jan, *A front-tracking method for the computations of multiphase flow*, J. Comput. Phys. **169** (2001), no. 2, 708–759, DOI 10.1006/jcph.2001.6726. MR3363449
[41] B. R. Vatti, *A generic solution to polygon clipping*, Communications of the ACM **35** (1992), no. 7, 56–63.
[42] O. Veblen, *An application of modular equations in analysis situs*, Ann. of Math. (2) **14** (1912/13), no. 1-4, 86–94, DOI 10.2307/1967604. MR1502443
[43] Q. Zhang, *On a family of unsplit advection algorithms for volume-of-fluid methods*, SIAM J. Numer. Anal. **51** (2013), no. 5, 2822–2850, DOI 10.1137/120897882. MR3116647
[44] Q. Zhang, *On donating regions: Lagrangian flux through a fixed curve*, SIAM Rev. **55** (2013), no. 3, 443–461, DOI 10.1137/100796406. MR3089409
[45] Q. Zhang, *On generalized donating regions: classifying Lagrangian fluxing particles through a fixed curve in the plane*, J. Math. Anal. Appl. **424** (2015), no. 2, 861–877, DOI 10.1016/j.jmaa.2014.11.043. MR3292705
[46] Q. Zhang, *GePUP: generic projection and unconstrained PPE for fourth-order solutions of the incompressible Navier-Stokes equations with no-slip boundary conditions*, J. Sci. Comput. **67** (2016), no. 3, 1134–1180, DOI 10.1007/s10915-015-0122-4. MR3493498
[47] Q. Zhang, *HFES: a height function method with explicit input and signed output for high-order estimations of curvature and unit vectors of planar curves*, SIAM J. Numer. Anal. **55** (2017), no. 2, 1024–1056, DOI 10.1137/15M105001X. MR3639580
[48] Q. Zhang, *Fourth- and higher-order interface tracking via mapping and adjusting regular semianalytic sets represented by cubic splines*, SIAM J. Sci. Comput. **40** (2018), no. 6, A3755–A3788, DOI 10.1137/17M1149328. MR3875807
[49] Q. Zhang and A. Fogelson, *MARS: an analytic framework of interface tracking via mapping and adjusting regular semialgebraic sets*, SIAM J. Numer. Anal. **54** (2016), no. 2, 530–560, DOI 10.1137/140966812. MR3470740

School of Mathematical Sciences, Zhejiang University, 38 Zheda Road, Hangzhou, Zhejiang Province, 310027 People's Republic China
Email address, Qinghai Zhang: qinghai@zju.edu.cn

School of Mathematical Sciences, Zhejiang University, 38 Zheda Road, Hangzhou, Zhejiang Province, 310027 People's Republic China

毕业论文（设计）文献综述和开题报告考核

对文献综述、外文翻译和开题报告评语及成绩评定：

成绩比例	文献综述 占（10%）	开题报告 占（15%）	外文翻译 占（5%）
分值			

开题报告答辩小组负责人（签名）_____

年　月　日