

非流形网格转化方法及其应用研究

孟文龙¹, 于航¹, 濮彦博¹, 史晓晓¹, 辛士庆²

(1. 哈尔滨工业大学(威海) 计算机科学与技术学院, 山东 威海 264209;

2. 山东大学 计算机科学与技术学院, 山东 青岛 266237)

摘要: 计算机图形学中的许多建模操作会产生非流形曲面, 虽然非流形曲面具有更加复杂的拓扑特性和更强大的几何描述能力, 但图形学领域中许多网格处理算法包括骨架提取、网格简化、细分等都要求输入的网格具有二流形的性质。为了确保已生成的非流形曲面能够与现有图形学算法兼容, 提出一种方法, 利用改进的双链表数据结构(DLFL)将非流形曲面转化为在几何外观上近似的二流形拓扑结构, 以此构建非流形网格曲面与传统数字几何处理之间的桥梁。为了验证算法框架的通用性, 将其应用于非流形曲面的3个关键应用场景: 测地距离场计算、网格简化和最远点采样。通过对这些应用实例的深入分析, 验证了算法在不同场景下的稳健性和准确性。实验结果表明, 算法在各应用场景中表现出显著成效, 进一步证实了其在实际应用中的潜力。

关键词: 非流形曲面; 双链表; 测地距离场; 网格简化; 最远点采样

DOI: 10.11907/rjdk.241791

开放科学(资源服务)标识码(OSID):



中图分类号: TP391

文献标识码: A

文章编号: 1672-7800(2025)001-0093-09

Research on the Conversion Method of Non-Manifold Meshes and Its Applications

MENG Wenlong¹, YU Hang¹, PU Yanbo¹, SHI Xiaoxiao¹, XIN Shiqing²

(1. School of Computer Science and Technology, Harbin Institute of Technology, Weihai 264209, China;

2. School of Computer Science and Technology, Shandong University, Qingdao 266237, China)

Abstract: In computer graphics, many modeling operations result in non-manifold surfaces. Although non-manifold surfaces have more complex topological properties and a stronger geometric description capability, many mesh processing algorithms in the field of graphics, including mesh simplification and subdivision, require the input meshes to possess the property of being two-manifold. Therefore, to ensure compatibility with existing graphics algorithms, this paper proposes a method for converting non-manifold surfaces into geo-metrically similar manifold topology structures. This method aims to bridge the gap between non-manifold mesh surfaces and traditional digital geometry processing techniques. To demonstrate the universality of the algorithm framework, this paper applies it to three key application scenarios involving non-manifold surfaces: including the computation of geodesic distance fields, mesh simplification and farthest point sampling. Through an in-depth analysis of these application examples, the robustness and accuracy of the algorithm in different scenarios are verified. Experimental results demonstrate that the algorithm exhibits significant effectiveness in each application scenario, further confirming its potential in practical applications.

Key Words: non-manifold surfaces; double linked face list; geodesic field; mesh simplification; farthest point sampling

收稿日期: 2024-09-12

扫描二维码阅读全文:



基金项目: 国家自然科学基金项目(62302124); 山东省自然科学基金项目(ZR2023QF122); 哈尔滨工业大学青年教师发展基金项目(IDGA10002143)

作者简介: 孟文龙(1992-), 男, CCF会员, 哈尔滨工业大学(威海)计算机科学与技术学院讲师, 研究方向为计算机图形学; 于航(2004-), 男, 哈尔滨工业大学(威海)计算机科学与技术学院学生, 研究方向为计算机图形学; 濮彦博(2004-), 男, 哈尔滨工业大学(威海)计算机科学与技术学院学生, 研究方向为机器人路径规划; 史晓晓(2005-), 女, 哈尔滨工业大学(威海)计算机科学与技术学院学生, 研究方向为计算机图形学; 辛士庆(1979-), 男, 博士, 山东大学计算机科学与技术学院教授, 研究方向为计算机图形学、计算几何。本文通讯作者: 孟文龙。

0 引言

数学领域中,流形(Manifold)是一种用于刻画几何对象内蕴拓扑结构的概念^[1-2]。具体而言,一个 N 维拓扑流形在其每一点的局部邻域都与 N 维欧几里得空间(Euclidean Space)存在同胚的一一对应关系,这种关系将局部区域映射到欧氏空间,局部同胚映射共同构建了整个流形结构。在计算机图形学领域,二流形(2-Manifold)曲面(见图1)往往被视为标准处理对象。基于二流形的特性,众多关键的图形学算法得以蓬勃发展,其中包括三维网格简化^[3-4]、多边形网格细分^[5]以及离散测地距离场计算^[6]等。这些算法以其严谨性、快速性和高效性而备受青睐,在处理标准二流形曲面时表现出色。

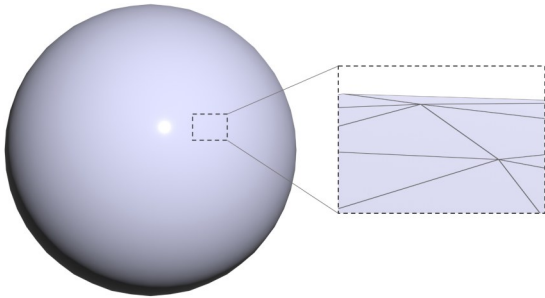


Fig. 1 When observing a three-dimensional sphere in a 2-manifold space, it exhibits locally plane-like characteristics

图1 在二流形空间中观察三维球体时局部呈现出类似平面的特性

但是,上述算法普遍将非流形曲面视作不符合预期的异常情况,处理方式往往采取错误提示或触发报错机制。对于诸如集合运算^[7-8]、融合处理^[9]和塌陷操作^[10-11]等基础建模任务,在执行时并不总能保持封闭性。这意味着,这些操作的输出并不总是有效的二流形曲面。因为许多图形算法都是预设工作在良好定义的流形曲面之上,当这种情况发生时,可能导致随后的一系列图形处理步骤无法得到预期的正确结果。因此,在没有确保操作结果仍保持流形结构的情况下,即使是精心设计的算法也可能面临失效的风险。本文从离散化角度出发,提出了一套完整的非流形三角网格曲面转化算法框架。该框架能够将输入的非流形结构高效地转化为在几何外观上近似的二流形拓扑结构,使得各种基于二流形曲面的图形学算法得以直接应用,简化了处理流程并提升了计算效率。

1 相关工作

目前,非流形网格修复算法主要可以分为两大类^[12]:全局修复法和局部修复法。全局修复法致力于一次性解决一系列与非流形相关的问题,并确保最终输出是严格的二流形网格曲面;而局部修复法则专注于检测并修复某一特定的非流形问题。

(1)全局修复法。大多数全局方法采用体素作为中间

表示以实现算法功能。这其中的典型方法包括体素网格占用法^[13-14],将体素网格编码为表征距离边界的距离函数法^[15-17],以及应用BSP(Binary Space Partitioning)树对目标体积进行非规则镶嵌^[18-19]等技术。这些方法通过对体素空间进行标记,以便明确区分模型的内部与外部区域,并提取出清晰的流形曲面作为最终结果。尽管全局方法确保了输出结果的流形性质,但它们在处理过程中通常需要对输入的网格进行重采样和重新网格化,这可能是由于要将数据转换为规则网格的需求,或者是因为使用了BSP树进行空间的重新镶嵌。这种处理方式在某些情况下并不理想,尤其是当输入网格具有精细的尖锐特征或已经携带了如纹理贴图额外数据时,这些操作可能会损害模型的原有细节和附加信息。

(2)局部修复法。不同于依赖体素表示的全局修复方法,局部修复技术通常直接对网格本身进行操作,从而避免了对输入数据的不必要改变^[20-21]。例如,Guéziec等^[22]提出了一种局部修复方法,该方法首先在非流形网格模型中的奇异顶点和边处进行分解,随后在确保流形性的前提下,通过贪婪算法合并各部分,最终将其转换为满足流形性质的网格。Attene等^[23]在修复实体对象的非流形网格时,仅专注于检测并修复存在错误的区域。此外,为了解决网格的自相交问题,他们采用了一种混合几何内核^[24-25]以细分自交的网格面片,进而得到一个非流形复形。然后,通过重复的区域生长过程从该复形中提取最外层的面构成流形外壳。从本质上讲,本文所提出的算法也是一种针对非流形网格的局部修补技术。该算法只专注于对非流形边进行有效修改,以将其转换为流形网格,从而确保网格的完整性和一致性。

2 背景知识

2.1 流形与非流形

流形是局部类欧氏空间,其连续性和可微性适用于任何形体描述;非流形则不满足这些条件,常存在共享边或顶点的多个面且处理复杂。在三维建模和图形学中,流形简洁易用,非流形虽然提供更灵活的表现力,但往往需更复杂的算法支持。

定义1 n -流形^[26-27]。设 H 是Hausdorff空间,若在 H 中任取一点 p ,总可以找到一个邻域 N ,该邻域与 n 维欧几里得空间 \mathbb{R}^n 的开集是同胚的,则称 H 为一个 n 维拓扑流形。

n -流形是对局部具有欧氏空间概念的推广。引入流形的概念源于人们在研究曲面时追求其拓扑结构的优良性。流形可以被看作是对曲面“柔软度”和可塑性的数学描述,它允许在不失一般性的前提下,将复杂的几何形状简化为易于处理的局部欧氏结构。

定义2 2-流形^[28]。如果一个表面上的任意点都拥

有一个足够小的邻域, 这个邻域与平面上的圆盘拓扑同胚。对于有边界的曲面, 那么边界上的点存在足够小的邻域与平面上的半圆盘同胚。那么, 这样的曲面被称为二流形曲面。如果不满足这些条件, 则被称为非二流形曲面。

在由点、线、面构成的三角网格曲面中, 可以这样精炼地定义二流形表面的特性: ①每条边仅被两个三角面共享, 以确保网格的封闭性; ②与每个顶点相邻接的边和面恰好组成一个闭环, 该特性保证了顶点局部连续性。上述两个条件分别从边和顶点的角度阐述了二流形三角网格曲面的定义。具体而言, 若一条边 l 被超过两个三角面共享, 则该边被视为非流形边, 如图 2(a) 所示 (彩图扫 OSID 码可见, 下同)。类似地, 如果网格上一个顶点 v 连接的边和面形成了两个或更多与圆盘拓扑同构的“环”结构, 这样的顶点被称作非流形顶点, 如图 2(b) 所示。只有当三角网格的所有顶点和边都不具备非流形特征时, 该网格才能被认定为二流形网格。若存在任何非流形的顶点或边, 则该网格被判定为非流形网格。图 3 展示了依照上述非流形定义所识别出的非流形网格中非流形边结构。

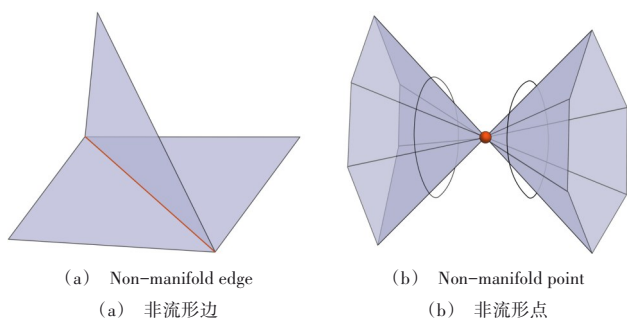


Fig. 2 Common non-manifold structures on triangular mesh surfaces

图 2 三角网格曲面上常见的非流形结构

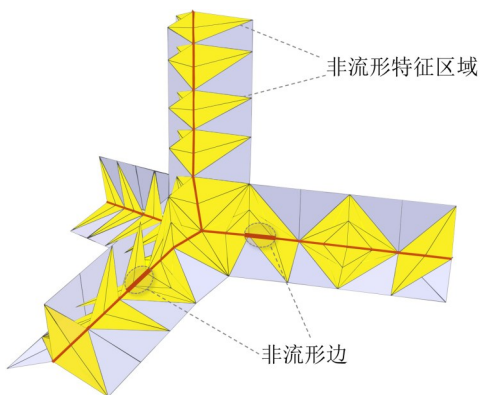


Fig. 3 Non-manifold regions and non-manifold edges on non-manifold mesh

图 3 非流形网格上非流形区域与非流形边

2.2 网格曲面表示的数据结构

边界表示方法 (Boundary Representation, B-rep) 能够全面提供实体的边界信息, 详细描述了模型的拓扑组件, 如面、边和顶点, 以及它们之间的连接关系^[29]。同时, 还涵盖了这些组件的几何定义, 包括面、线和点的具体形状和位置。边界表示法包含多种数据结构, 用以描述形体的边

界信息。其中, 翼边结构 (Winged Edge Structure)、半边结构 (Halfedge Structure) 是常见的表示方式。

(1) 翼边结构^[30]。翼型边线结构是一种基于边结点的列表, 其核心在于形体的边。通过边作为起点, 可以迅速查找到与该边相关联的所有面、邻边、端点以及其他相关信息。这种数据结构的显著优点是它对于拓扑查询和局部操作的高效支持。图 4 展示了翼边结构的使用示例。然而, 翼边数据结构主要适用于流形几何实体的表示, 对于非流形几何实体, 它无法完整地呈现实体的拓扑信息。

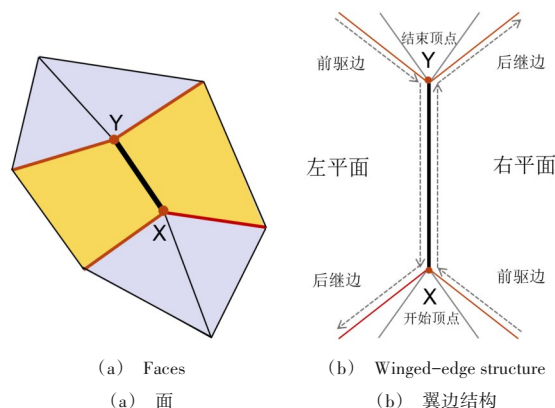


Fig. 4 The winged-edge structure of a vertex on a triangular mesh

图 4 三角网格面上一个边结点的翼边结构

(2) 半边结构^[31]。半边数据结构是在翼边数据结构的基础上发展而来。在这种数据结构中, 翼边数据结构中的每条整边被细分为两条半边, 每条半边都以其所连接的顶点为起点。通过这种细分, 半边数据结构得以通过体 (Solid)、面 (Face)、环 (Loop)、半边 (Half-edge) 和顶点 (Vertex) 5 个层次的拓扑结构以精确地描述实体。传统的半边结构在设计时并未考虑存储一点处周围半边的数据, 这种局限性使得半边结构无法妥善处理非流形。

基于边的数据结构在稳定性方面表现出色, 能够有效地避免错误方向的多边形、重叠多边形等问题。然而, 这类数据结构通常不适用于描述非流形网格曲面, 特别是在处理包含非流形边的场景时。鉴于此, 本文选择了以面为基础的数据结构——双链面表 (Double Linked Face List, DLFL), 以便更好地应对非流形表面的建模需求^[27, 32]。双链面表 (DLFL) 数据结构由 3 个核心部分构成: 面列表 F (Face List)、边列表 E (Edge List) 和顶点列表 V (Vertex List)。每个面节点由一系列首尾相连的有向边组成, 这些有向边形成了一个单向循环链表, 刻画面的封闭边界; 边列表 E 的每个边节点则拥有两个指向其面节点的双向指针; 顶点列表 V 中的每个顶点节点都设有一个指针数组, 这些指针直接映射到包含该顶点面节点 F 中的对应位置。这种设计允许快速访问和操作与边相关的面信息, 四面体对应的双链面表数据结构如图 5 所示。

DLFL 数据结构在计算几何中尤为重要, 特别是在处理多边形和多面体的几何运算时。它不仅能够高效地表

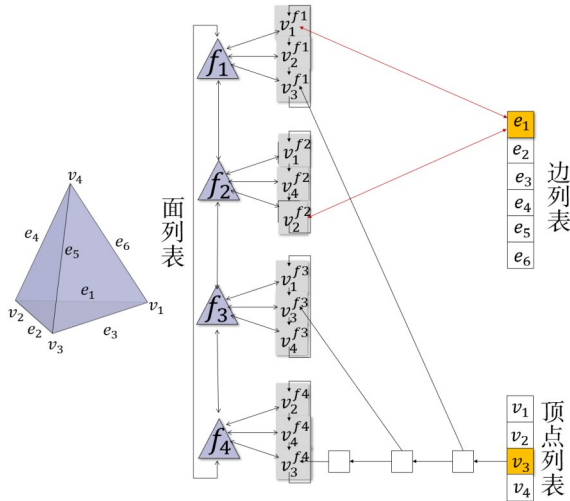


图5 The DLFL representation of a tetrahedron

图5 一个四面体的双面链表表示

示和存储复杂的多面体结构,还能支持多种几何操作,如插入、删除、分割和合并面等。通过维持面、边和顶点之间的双向链接,DLFL数据结构能够在恒定时间内完成许多基本操作,从而显著提高算法整体效率。

3 非流形网格转化方法

3.1 非流形网格数据结构—改进的双链表

双链表数据结构提供了结点与面($v \rightarrow f$)、边与面($e \rightarrow f$)以及边与结点($e \rightarrow v$)之间的多种拓扑关系,为建模操作提供了强大支持。具体而言,借助顶点列表V,能够迅速定位到与特定顶点相邻的所有面。同样,利用边列表E,也能在极短的时间内找到与某条边相邻的所有面集。这种高效的数据检索能力极大提升了建模操作的便捷性。然而,这种数据结构是专为刻画二流形网格曲面设计,当输入数据具有非流形特性时,利用双链表进行描述可能会遇到困难。在传统的边列表数据结构中,每个边节点通常配备两个双向指针,分别指向两条与之相连的有向边。但是,对于某些非流形结构,比如存在由多个面共享边的情况,这样的边节点需要包含超过两个以上指针,以反映它所连接多于两条边^[27]。这导致了传统双链表数据结构无法有效表示这类非流形几何特征,因而需要对数据结构进行相应的扩展或调整。

为了充分表示和处理所有可见的非流形结构,本文对传统的双链表数据结构进行了改进。在经过改进的数据结构中,边列表E中的每个边节点现在包含一个扩展的双向指针集合,这些指针分别指向与该边相邻的面。指针集合的大小直接反映了与该边相邻接的面的数量。通过这种方式,能够有效地描述和处理非流形边。图6展示了非流形结构的双链表表示。

改进后的双链表数据结构不仅保留了原有丰富的拓扑信息,还通过新增的边对面连接,极大地便利了对网

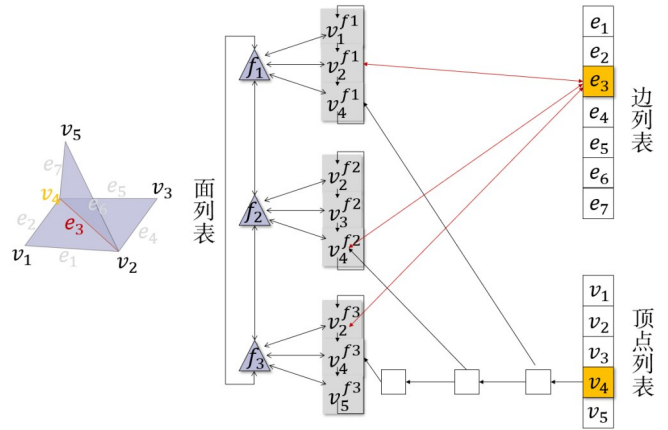


图6 An improved DLFL representation of a non-manifold structure

图6 一个非流形结构的改进双链表表示

格的处理。这种增强使在识别和标记非流形边时更为高效。此外,该结构简化了在随后的非流形边转换算法中的关键步骤,如顶点和边的复制,以及新顶点/边的插入等操作。采用双链表的方法,显著降低了这些操作的时间复杂度,提高了整体处理效率。

3.2 双重覆盖生成二流形网格

本文将阐述双重覆盖生成二流形网格的核心步骤。为了便于理解,本文将一个由不同顶点构成的三角网格曲面作为输入进行说明。值得注意的是,本文构建方法在原则上适用于任意三角剖分(从形式上而言,适用于任何纯粹的二维 Δ -复形)。对于输入的非流形三角网格曲面M,采用边作为分类基准。利用双链表数据结构,将网格中的边明确区分为流形边和非流形边两个主要类别。针对非流形边,执行以下步骤:

Step1: 对于任意非流形边 e_m ,设其端点为 a 和 b 。先确定与顶点 a 相邻的面集合 $\{f_i | i = 1, 2, \dots, k\}$ 。再根据“可到达性”关系对集合 $\{f_i | i = 1, 2, \dots, k\}$ 进行划分。构建此划分的过程中,如果两个相邻的面 f_1 和 f_2 被归入同一子集中,则认为它们满足“可到达性”条件。该条件成立的充要条件是:面 f_1 和 f_2 具有一条公共边,且这条边是与顶点 a 相邻的流形边。

Step2: 一旦确定了子集的数量为 s ,需要对顶点 a 和 b 执行 s 次复制操作。具体而言,将顶点 a 的属性,包括其几何、纹理等坐标一一复制到所有 s 个副本中。这些副本的坐标属性与原顶点 a 保持一致,但它们各自拥有唯一的标识符。

Step3: 对于每个创建的副本,赋予它们 $0 \sim (s - 1)$ 的编号。在所有与顶点 a 邻接的面中更新顶点信息,用新分配的顶点编号替换掉原有顶点 a 的编号,以完成顶点复制操作。

对于流形边 $e_m = ab$ 而言,其操作相对简单。因为根据定义,流形边 e_m 连接的两个邻接面 f_1 和 f_2 是“可到达的”。因此,只需对 e_m 的两个顶点 a 和 b 进行两次复制操

作,并对产生的副本进行编号,便完成了复制过程。

Step4:选择从某个网格三角形出发,执行向外的泛洪扫描(Flood-Fill Scanning)以更新重建双链面表结构。这一过程涉及重复之前的Step1至Step3,确保对所有流形和非流形边进行处理,并正确更新所有相关面的邻接性。通过这一系列操作,将获得消除了非流形特征的双重覆盖的二流形网格曲面 M^* ,用于后续几何处理。

算法 1 双重覆盖生成二流形网格

输入:非流形三角网格 M

输出:二流形三角网格 M^*

1. FOR 非流形网格 M DO
2. 识别 M 中非流形面 f_n 及对应非流形边 e_n
3. END
4. 选择从某个网格三角形出发,执行向外的泛洪扫描以更新重建双链面表结构,并执行下面步骤:
5. FOR 每个非流形边 e_n DO
6. 确定相邻面集合 $\{f_i | i = 1, 2, \dots, k\}$, 根据“可达性”划分为 s 个子集。
7. e_n 对应的顶点 a 和 b 执行 s 次复制操作,创建 s 个副本。对每个创建的副本赋予它们 $0 \sim (s - 1)$ 编号,完成顶点复制操作。
8. END
9. FOR 流形边 e_m DO
10. 顶点 a 和 b 进行两次复制操作,并对产生的副本进行编号。
11. END

为了提高数据结构的效能并优化存储管理,在复制原始网格模型 M 上的顶点 v 时,本文采用了一种高效的映射策略。具体而言,对于 M 上的每一顶点 v ,在增强后的网格模型 M^* 中可能会有两个或多个对应的顶点。为了维持 M 与 M^* 之间的一致性,并且有效降低存储需求,仅记录新增顶点所对应原始顶点的标识符,而不是它们详细的坐标、材质等信息。

图 7 展示了非流形网格曲面 M 使用上述策略转换为流形网格 M^* 的可视化结果。采用双重覆盖方法将非流形网格转换为二流形网格,使得现有图形学算法无需修改即可用于计算和处理原本不兼容的非流形网格,极大扩展了这些算法的适用范围,并简化了复杂网格结构的处理流程。

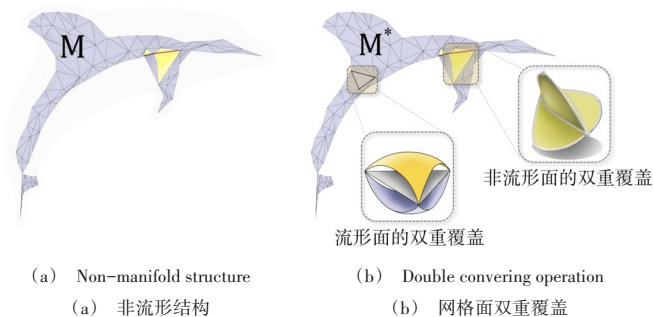


Fig. 7 Converting a non-manifold mesh using double-covering method

图 7 双重覆盖方法转换非流形网格过程

3.3 实验结果

本文实验基于 Windows 10 操作系统,并采用 C++ 语言对算法加以实现。为了保证一致性和可重复性,在一台配备 Intel i7-12700KF CPU 的计算机上进行了所有实验,CPU 运行频率为 3.60 GHz,内存为 16 GB。为了展示本文方法的性能,采用了 4 个基准模型,如图 8 所示。可以看出,这些模型表现出了不同的复杂度,包括网格复杂度、拓扑复杂度、非流形结构复杂度等。

对于每个基准模型测试,本文算法与 Fei 等^[33]提出的先进非流形网格处理方法进行比较。该方法通过剪切、缝合等一系列几何操作生成拓扑划分和连接上的流形网格结构。表 1 展示了在每个基准模型上的时间和内存消耗情况。

Table 1 Comparison of results under different baseline models

表 1 不同基准模型下计算结果比较

基准模型	本文方法		Fei 等 ^[33] 方法	
	时间(ms)	内存(MB)	时间(ms)	内存(MB)
图 8(a)	0.8	0.073	9	0.104
图 8(b)	11	0.576	79	0.717
图 8(c)	35	1.137	211	1.856
图 8(d)	62	1.922	594	2.518

实验结果表明,与 Fei 等^[33]的方法相比,本文算法表现出了优异性能。例如,在如图 8 所示的第 3 个基准模型中,本文方法实现只需要 35 ms 和 0.576 MB 内存即可完成非流形网格的转化操作,而对比方法需要 211 ms 和 0.717 MB 内存。由此可知,本文方法在时间和空间等指标上优势明显。

本文算法在应用于不同分辨率和三角剖分质量的非流形网格时表现出鲁棒性。为了验证该特征,在具有不同分辨率和质量的非流形网格上进行转化。通常评估网格质量时会使用 Lo 值作为通用度量标准, Lo 值越高表示网格质量越接近最优^[34]。图 9 展示了 4 种不同分辨率(分别为 0.2 K 面、2 K 面、20 K 面、200 K 面)和 4 种不同网格质量(对应的 Lo 值分别为 0.56、0.42、0.27 和 0.11)的模型上执行算法框架以实现非流形网格的转化。

图 10 展示了本文算法生成结果的时间和内存表现。该方法并没有因输入网格复杂度的升高和质量的降低而导致时间和内存等指标的急剧增长,证明了本文方法能够高效鲁棒地产生可靠的流形网格结果。

4 算法应用

本文将通过解决图形学中的 3 个关键问题——测地距离场计算、网格精简和最远点采样以阐述本文算法在转化非流形网格后的重要应用价值。

4.1 测地距离场计算

测地线在微分几何领域占据着重要地位,它定义了曲面上两点间距离最短的路径,可以类比于二维平面上连接

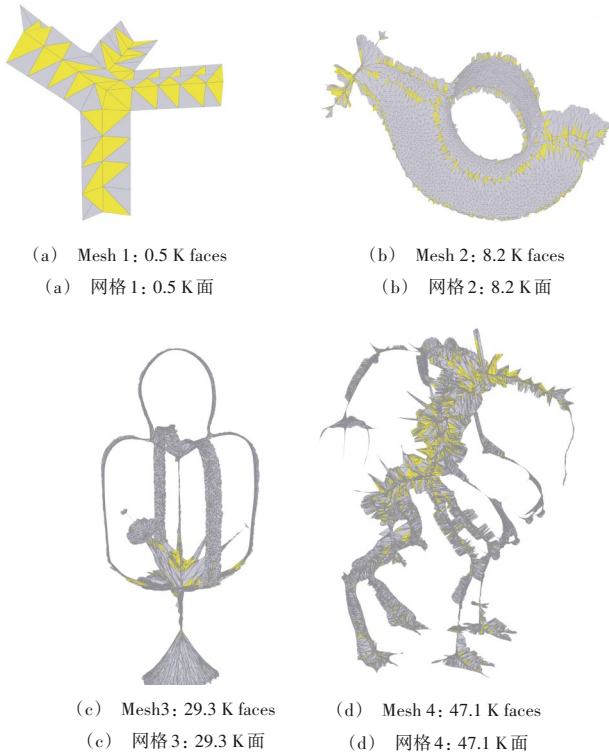


Fig. 8 Non-manifold meshes with varying levels of complexity
图8 复杂程度不同的非流形网格

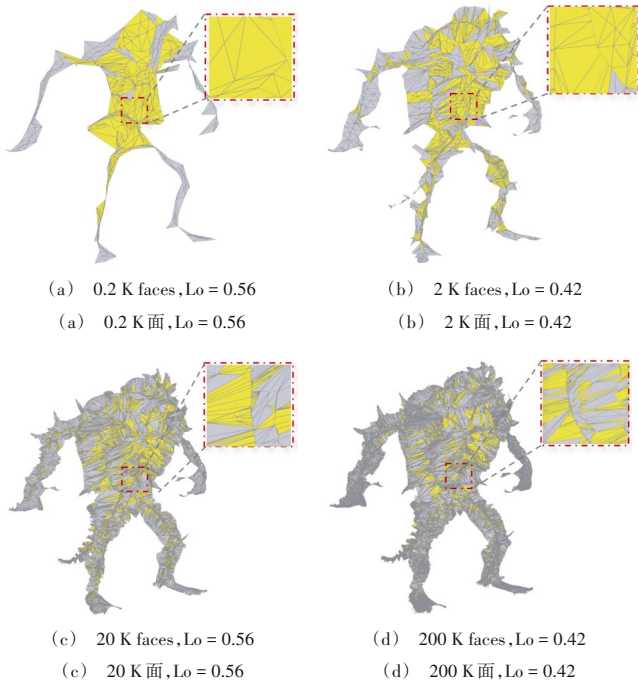


Fig. 9 Non-manifold meshes with different resolutions and Lo quality

图9 不同分辨率和Lo质量的非流形网格展示

两定点的直线段。这一概念不仅在数学理论中具有基础性作用,而且其应用价值也极为广泛,涉及计算机图形学、图像处理、计算几何和计算机视觉等多个领域。自20世纪80年代起,离散测地线的研究取得了显著进展,众多学

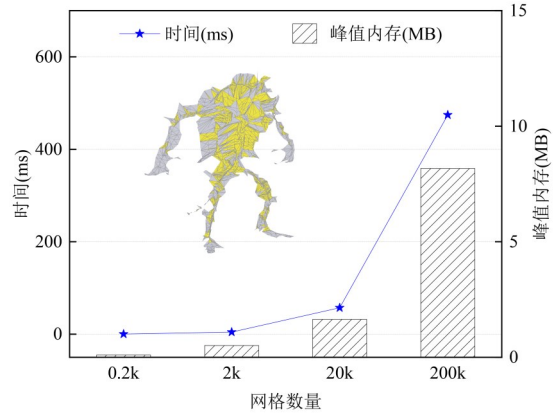


Fig. 10 The time and memory performance of the algorithm

图10 本文算法生成结果的时间和内存表现

者致力于这一领域的探索,并提出了多种实用性强的计算方法。在网格曲面上MMP^[35](Mitchell-Mount-Papadimitriou)算法和CH^[36](Chen-Han)算法被广泛认可。基于这两种算法,研究者还发展了若干改进型算法^[37-39]。在众多离散测地线算法中,VTP(Vertex-Oriented Triangle Propagation)算法因其优异的时空效率而备受关注^[39]。该算法探讨了当两个窗口同时抵达同一网格三角形时的场景,并针对此情境提出了一整套系统的裁剪准则,是无需预处理便能计算精确测地线距离的最新技术。然而,上述算法通常只适用于二维流形网格曲面。本文提出的非流形转化算法可以成功地将VTP算法与非流形网格曲面相结合,实现了二者之间的技术桥接。

依据本文所述的非流形网格转换方法,可以成功地将原始非流形网格转化成一个双层的流形网格曲面M*。虽然M*是双层结构,但它在局部范围内保持了与原曲面M的等距特性。根据共形几何理论中的一个关键性质,即“测地路径可以被理解为共形参数平面上以共形因子为权值的最优路径”^[40],可以得到以下关于网格顶点v₁和v₂之间实际测地距离的推导:假设顶点v₁产生了两份拷贝v₁⁽¹⁾和v₁⁽²⁾,顶点v₂也产生了两份拷贝v₂⁽¹⁾和v₂⁽²⁾。根据上述论断,可以认为v₁与v₂之间的实际测地距离是在所有可能路径中具有最小共形因子加权的那条路径的长度,如式(1)所示。

$$g(v_1, v_2) = \min \{ g(v_1^{(1)}, v_2^{(1)}), g(v_1^{(1)}, v_2^{(2)}), g(v_1^{(2)}, v_2^{(1)}), g(v_1^{(2)}, v_2^{(2)}) \} \quad (1)$$

其中,g(·)表示测地距离。根据式(1),使用VTP算法^[39]计算转化后非流形曲面上的测地路径和测地距离场,如图11所示。此外,通过对转化后的具有不同顶点数量的非流形网格进行实验,并采用随机取样的方法,获得了测地距离场的平均计算时间与网格中三角形数量的关系,如表2所示。

处理非流形曲面时,一种简洁而直观的方法是识别并移除非流形特征,以消除这些结构带来的问题。随后,在修正后的模型上应用标准的测地线算法以计算测地距离。

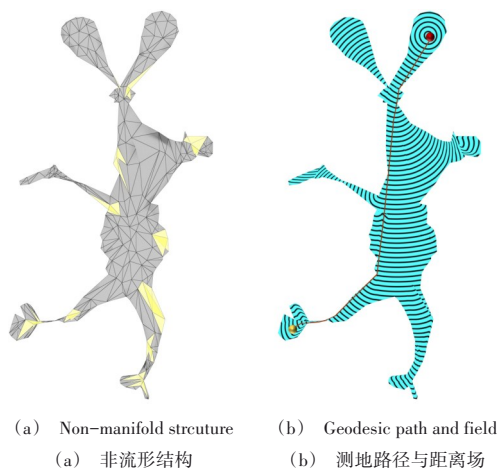



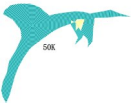
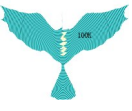
Fig. 11 Geodesic paths and distance fields on non-manifold meshes using the VTP algorithm

图 11 非流形网格上使用 VTP 算法计算测地路径及距离场

然而,从表 2 可以看出,本研究提出的转化方法能够以较低的预处理成本将非流形网格转换为满足流形要求的表示形式。这种转换允许无需额外修改即可直接利用现有的测地线算法进行计算,从而显著提高了处理非流形网格的效率和实用性。相比之下,去除非流形结构的处理方式由于在处理过程中移除了非流形部分,所得到的模型在局部是不完整的。因此,在这种经过修改的模型上进行计算,可能会得到不全面的结果,无法准确反映原始模型的全部特征,从而影响了结果的完整性和准确性。

Table 2 Statistics for calculating geodesics on non-manifold meshes

表 2 非流形网格上计算测地距离场的时间统计

时间/s			
预处理 t_p	0.004	0.051	0.113
测地距离场 t_g	0.071	0.685	1.580
预处理时间占比	0.056	0.069	0.067

4.2 网格简化

在计算机图形学和计算机视觉的众多应用中,经常需要处理具有高网格密度和精细度的三维模型,以精确地表示虚拟对象。然而,这些高细节层次的网格并不总是能直接用于渲染或模拟,因为其高精度要求会对 GPU 造成巨大的计算压力。因此,在保持原始网格外观的同时,使用尽可能精简的网格来表示它变得至关重要。这种网格优化的需求,促进了网格简化技术的发展。

QEM(Quadric Error Metrics)技术是一种先进的基于点对的网格压缩方法^[41]。该技术的核心在于采用一种二次误差度量,用以精确地确定新顶点的位置,从而实现高效的网格简化。在 QEM 的框架下,网格简化被视作一系列的边折叠(Edge Collapse)操作。其中,一条边的两个端点合并为一个新点。这个新点的位置是通过最小化一个二次误差函数确定,如式(2)所示。

$$v^* = \arg \min_v \sum_p dis(v, p)^2 \quad (2)$$

其中, $p \in (plane(v_1) \cup plane(v_2))$, $plane(v_i)$ 表示网格上的点对应原始三角面, v^* 是合并的新点。该函数衡量了合并操作对网格表面带来的畸变。通过这种方式, QEM 不仅能够有效地减少模型中的顶点数量,还能在保持视觉质量的同时,确保简化后模型的几何精度。标准 QEM 网格简化算法的设计基于半边数据结构,因而在处理包含非流形特征的网格时具有局限性。由于它不具备直接操作非流形几何的能力,这导致在简化过程中遇到非流形结构时无法被正确处理,从而可能导致简化流程失败。

本文提出的非流形转换策略,通过将网格中的非流形部分修复为标准的流形结构,巧妙地解决了这一挑战。图 12 展示了在处理非流形网格 M 时,若直接采用 QEM 算法简化,网格 M 中的非流形部分将无法被适当处理。相反,当对经过转换的网格 M* 应用 QEM 算法时,原本复杂的非流形结构区域将得到有效简化。

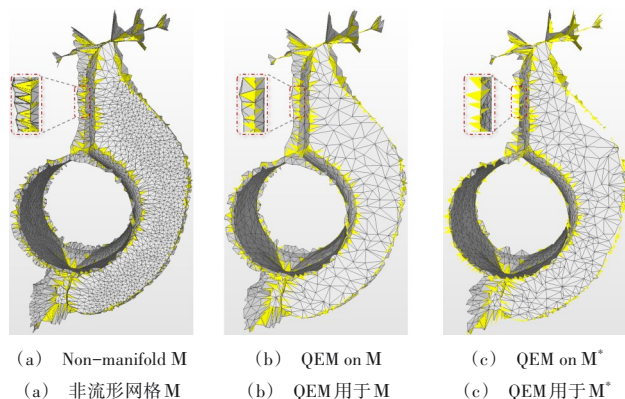


Fig. 12 Performance of the QEM algorithm applied directly to non-manifold meshes and to converted meshes in non-manifold regions

图 12 QEM 算法直接用于非流形网格和转化网格 M* 后在非流形区域的表现

4.3 最远点采样

与泊松采样不同,最远点采样(Farthest Point Sampling, FPS)是一种确保采样点尽可能分散的贪婪算法^[42]。Moenning 等^[43]将最远点采样的概念拓展到曲面上。在二维平面中,最远点采样通过迭代选择未采样区域中距已有采样点最远的点作为新的采样点,这个点通常是最大空圆(没有被其他采样点覆盖区域所对应的最大圆)的圆心。而在曲面上的最远点采样遵循相似的逻辑,主要区别在于对网格化的曲面,通常假设每个多边形单元内的测地距离场(考虑曲面内在几何的距离度量)是近似线性的。因此,在这种假设下,最远点通常位于多边形单元的顶点上。

在非流形曲面上实施经典最远点采样通常会遇到效果不佳的问题。然而,借助于本文提出的非流形曲面转换算法,能够在非流形网格曲面上计算出测地距离场。基于计算结果,可以按照以下步骤执行最远点采样过程:

Step1: 遍历所有顶点,确定具有最大距离值的顶点 v 。

Step2:以顶点 v 作为参考点,更新其他所有顶点的距离值,按照由近及远的顺序进行。

Step3:记录当前采样点 v 。检查已采样点的数量是否达到预设目标;若未达目标数量,则返回Step1继续执行;若已达目标数量,则终止算法。

图13展示了在转化后的非流形网格曲面上进行最远点采样,采集300个点的执行结果。

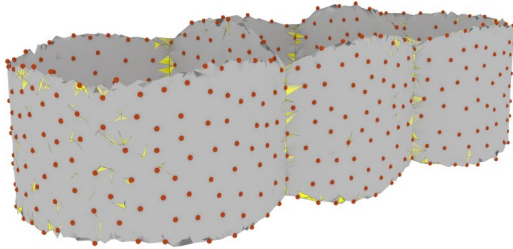


Fig. 13 Farthest point sampling on non-manifold meshes using the double covering strategy

图13 双重覆盖策略在非流形网格上执行最远点采样

5 结语

本文提出将非流形网格转化为在几何外观上近似的二流形拓扑结构框架,这一转化构建起非流形网格与传统数字几何处理算法之间的桥梁,为后续几何处理算法研究提供了新的视角和工具。为了全面评估该算法框架的有效性和适用性,将该框架应用于3个关键的非流形曲面应用场景:测地距离场计算、网格简化处理以及最远点采样技术。通过这些应用实例的深入分析,验证了该方法在不同场景下的稳健性和准确性。本文算法专注于非流形边的有效修改,确保网格的完整性和一致性,但对于含有非流形点及其复合形式的非流形网格无法有效解决。

未来,将进一步探索和研究适用于一般非流形网格曲面的通用转化框架。此外,将探索如何将目前数字几何处理算法直接应用于非流形网格的通用策略和方法。

参考文献:

- [1] SUN J G, HU S M. Fundamentals of computer graphics: a tutorial [M]. 2nd Edition. Beijing: Tsinghua University Press, 2009.
孙家广,胡事民. 计算机图形学基础教程[M]. 第2版. 北京:清华大学出版社,2009.
- [2] FOLEY J, ANDRIES V, STEVEN F. Computer graphics in C#: principles and practices[M]. Massachusetts: Wesley Publishing Company, 2008.
- [3] LIANG Y Q, FA Z H, XIAN T Z. 3D mesh simplification with feature preservation based on whale optimization algorithm and differential evolution [J]. Integrated Computer-Aided Engineering, 2020, 27 (4) : 417-435.
- [4] CAMPOMANES R, OSCAR C, SERGIO D. Evolutionary multi-objective optimization for mesh simplification of 3D open models [J]. Integrated Computer-Aided Engineering, 2013, 20(4): 375-390.
- [5] MA W Y, XI M, SHIU K T, et al. A direct approach for subdivision surface fitting from a dense triangle mesh[J]. Computer-Aided Design, 2004, 36 (6): 525-536.
- [6] DU J, YING H, ZHENG F, et al. On the vertex-oriented triangle propagation (VTP) algorithm: parallelization and approximation [J]. Computer-Aided Design, 2021, 130: 102943.
- [7] CHEN S X, MING C, SHENG L L, et al. A real time visual boolean operation on triangular mesh models [J]. Computer-Aided Design, 2022, 19 (3): 125-136.
- [8] WANG C L. Approximate boolean operations on large polyhedral solids with partial mesh reconstruction [J]. IEEE Transactions on Visualization Computer Graphics, 2010, 17(6): 836-849.
- [9] RISO M, GIACOMO N, ENRICO P, et al. Boolsurf: boolean operations on surfaces[J]. ACM Transactions on Graphics, 2022, 41(6): 1-13.
- [10] ALLIEZ P. Recent advances in compression of 3D meshes [C]//Proceedings of the 13th European Signal Processing Conference, 2005: 1-4.
- [11] HOPPE H. Meshes Progressive[Z]. Seminal Graphics Papers: Pushing the Boundaries, 2023, 2: 111-120.
- [12] ATTENE M, MARCEL C, LEIF K. Polygon mesh repairing: an application perspective[J]. ACM Computing Surveys, 2013, 45(2): 1-33.
- [13] JU T. Robust repair of polygonal models[J]. ACM Transactions on Graphics, 2004, 23(3): 888-895.
- [14] NOORUDDIN F S, TURK G. Simplification and repair of polygonal models using volumetric techniques [J]. IEEE Transactions on Visualization Computer Graphics, 2003, 9(2): 191-205.
- [15] CURLESS B L, MARC S L. A volumetric method for building complex models from range images [J]. Computer Graphics and Interactive Techniques, 1996, 5(2): 303-312.
- [16] HORNUNG A, LEIF K. Robust reconstruction of watertight 3D models from non-uniformly sampled point clouds without normal information [C]//Proceedings of the Eurographics Symposium on Geometry Processing, 2006: 23-40.
- [17] KAZHDAN M M, MATTHEW B, HUGUES H. Screened poisson surface reconstruction [C]//Proceedings of the 4th Eurographics Symposium on Geometry Processing, 2006: 234-246.
- [18] MURALI T M, THOMAS A F. Consistent solid and boundary representations from arbitrary polygonal data [C]//Proceedings of the Symposium on Interactive 3D Graphics, 1997: 23-30.
- [19] HU Y X, ZHOU Q, XI F G, et al. Tetrahedral meshing in the wild [J]. ACM Transactions on Graphics, 2018, 37(4): 1-14.
- [20] ROSSIGNAC J, DAVID E C. Manifold breps for non-manifold r-sets [C]//Proceedings of the Fifth ACM symposium on Solid Modeling and Applications, 2000: 12-21.
- [21] MARCO A. As-exact-as-possible repair of unprintable STL files [J]. Rapid Prototyping Journal, 2016, 12(2): 341-363.
- [22] GUEZIEC T L. Cutting and stitching: converting sets of polygons to manifold surfaces [J]. IEEE Transactions on Visualization Computer Graphics 2001, 7(2): 136-151.
- [23] ATTENE M. A lightweight approach to repairing digitized polygon meshes [J]. The Visual Computer, 2010, 26(11): 1393-1406.
- [24] ATTENE M. Direct repair of self-intersecting meshes [J]. Graphical Models, 2014, 76(6): 1-14.
- [25] ATTENE M. ImatiSTL-fast and reliable mesh processing with a hybrid

- kernel[M]. Berlin: Springer, 2017.
- [26] POPOVI J, HUGUES H. Progressive simplicial complexes[J]. *Computer Graphics*, 1997, 12(9): 212–230.
- [27] SUN M Z. Research on non-manifold surface transformation algorithms [D]. Changsha: Central South University, 2013.
孙漠舟. 非流形表面转化算法的研究[D]. 长沙: 中南大学, 2013.
- [28] REQUICHA A, VOELCKER H B. Solid modeling: current status and research directions [J]. *IEEE Computer Graphics Applications*, 1983, 3(7): 12–24.
- [29] TAI Z W, ZHANG X L, YIN X L, et al. Conversion of boundary representation of three-dimensional ship models to constructive representation methods [J]. *Journal of Computer-Aided Design & Computer Graphics*, 2024, 35(12): 1851–1862.
太志伟, 张兴龙, 尹晓龙, 等. 三维船舶模型边界表示转构造表示方法[J]. *计算机辅助设计与图形学学报*, 2024, 35(12): 1851–1862.
- [30] BAUMGART B G, WINGED E. Polyhedron representation [D]. California: Stanford University, 1972.
- [31] ZHANG C L, ZHU L, WEN S S, et al. Research on a unified data structure for representing manifold and non-manifold shapes [J]. *Aerospace Shanghai*, 2017, 34(4): 6–15.
张成林, 朱琳, 文珊珊, 等. 一种表示流形形体和非流形形体的统一数据结构研究[J]. *上海航天*, 2017, 34(4): 6–15.
- [32] ERGUN A, CHEN J. Guaranteeing the 2-manifold property for meshes with doubly linked face list [J]. *International Journal of Shape Modeling*, 1999, 5(2): 159–177.
- [33] FEI Y P, SONG Q C, DAN S, et al. A new algorithm for repairing non-manifold surfaces [C]// *Proceedings of the IEEE 10th International Conference on High Performance Computing and Communications*, 2023: 1704–1708.
- [34] LO S H. A new mesh generation scheme for arbitrary planar domains [J]. *International Journal for Numerical Methods in Engineering*, 1985, 21(8): 1403–1426.
- [35] MITCHELL S B, DAVID M M, CHRISTOS H. The discrete geodesic problem [J]. *SIAM Journal on Computing*, 1987, 16(4): 647–668.
- [36] CHEN J D, YI J H. Shortest paths on a polyhedron [C]// *Proceedings of the 6-th annual symposium on Computational Geometry*, 1990: 12–23.
- [37] XIN S Q, GUO J W. Improving Chen and Han's algorithm on the discrete geodesic problem [J]. *ACM Transactions on Graphics*, 2009, 28(4): 1–8.
- [38] TRETTNER P D, BOMMES L K. Geodesic distance computation via virtual source propagation [J]. *Computer Graphics Forum*, 2021, 40(5): 247–260.
- [39] QIN Y P, XIAO H, HONG C Y, et al. Fast and exact discrete geodesic computation based on triangle-oriented wavefront propagation [J]. *ACM Transactions on Graphics*, 2016, 35(4): 125–134.
- [40] AFLALO Y, RON K. Measuring geodesic distances via the uniformization theorem [C]// *Proceedings of the International Conference on Scale Space and Variational Methods in Computer Vision*, 2011: 12–23.
- [41] GARLAND M, HPAUL S. Surface simplification using quadric error metrics [C]// *Proceedings of the International Conference on Computer Graphics and Interactive Techniques*, 1997: 23–41.
- [42] KAMOUSHI P, SYLVAIN L, ANIL M, et al. Analysis of farthest point sampling for approximating geodesics in a graph [J]. *Computational Geometry*, 2016, 57: 1–7.
- [43] CAMBRIDGE C F, DNEIL N A, MOENNING C. Fast marching farthest point sampling [J]. *Eurographics*, 2003, 4: 1–15.

(责任编辑:孙娟)