

CAD网格模型的逆向CSG建模

杨全欧¹, 李志辉^{2*}, 贾晓红³, 石卫波⁴, 严冬明⁵

(1. 贵州理工学院 航空航天工程学院, 贵州贵阳 550003;

2. 北京流体动力科学研究中心, 北京 100011;

3. 中国科学院 数学与系统科学研究院, 北京 100190;

4. 中国空气动力研究与发展中心 超高速空气动力研究所, 四川绵阳 621000;

5. 中国科学院 自动化研究所, 北京 100190)

摘要: 针对复杂机械构型的3D网格难以全自动重建CSG(Constructive Solid Geometry)模型的问题, 文中提出了一种半自动交互式CAD(Computer Aided Design)网格模型重建CSG方法. 制定了布尔运算的模型命名规则. 首先采用适度用户交互的半自动分割方法对网格模型进行分割, 以布尔树结构将分割模型导出并相应命名, 获得批量具备几何基元特征的网格文件; 改进基元拟合函数, 建立了适于命名规则的自动拟合平台, 对网格文件批量拟合, 获得可供CSG生成的中间文本文件; 基于OCC(Open CASCADE)构建了以中间文本为输入的基元生成与布尔运算系统, 实现了CSG模型的自动生成. 在类天宫飞行器复杂网格模型上进行CSG重建, 获得了模型的绝大部分细节, 具备与原模型高度吻合的表面特性. 在自动方法难以处理的穿孔模型上, 成功重建出实体模型.

关键词: 网格分割; 重建CSG; 命名规则; 基元拟合; 布尔运算

中图分类号: TP391.41

文献标识码: A **文章编号:** 1000-4424(2025)04-0394-17

§1 引言

航天器再入大气过程面临强气动力热环境, 多场耦合力学计算是数值预报问题的重难点^[1-2], 其中热力响应导致的结构外形变化是外流场计算误差之一. 从结构网格或点云构造实体几何(constructive solid geometry, CSG)可作为多场耦合计算中结构外形变化的模型更新方案, 在原始设计文件丢失或实物模型不可用的情况下发挥着重要作用^[3], 也是计算机图形学的基本任务^[4]. 然而, 航天器等高复杂度机械产品的CAD网格模型通常由多种规则与自由曲面混合构

收稿日期: 2023-07-09 修回日期: 2025-10-09

*通讯作者, E-mail: zhli0097@x263.net

成,其几何与拓扑关系极为复杂.当前,基于深度学习的全自动重建方法严重依赖于大规模标注数据集,且在处理此类具有精确工程语义的复杂构型时,往往难以保证重建模型的几何精度与布尔结构的正确性,导致其结果难以直接用于高精度力学计算等下游工程分析.传统的自动分割与拟合算法在缺乏高层语义引导的情况下,容易在模型细节特征处产生过分割或欠分割,并因误差累积导致后续布尔运算失败.因此,开发一种能够兼顾复杂模型重建精度与工程实用性的方法显得尤为重要.

CSG树记录了模型生成过程的基元集合结构,从网格或点云获得CSG实体是对树结构的逆向求解. Wu等^[5]直接在原始点云上构建CSG模型和树,将所有基元进行了重新拟合,在拉伸体上取得一定效果. Rabbani等^[6]基于CAD模型库,利用点云的约束构建模型,而现实中复杂模型需依赖于庞大的库. Ye等^[7]从扫描数据中构建基于特征的参数模型,这使网格与点云数据的原始模型得到重构. Ren等^[8]采用CSG的三级表示方法,以无监督方式的端到端训练实现点云到模型重建.

含装配外形的几何实体是力学计算等下游工程应用的输入,而复杂装配关系的网格模型通过自动重建获得保证基元属性精度的单实体仍是难题.因此,本文提出了一种半自动交互式CAD网格模型的CSG重构方法.该方法的核心思想是引入适度的用户交互,以弥补全自动方法在语义理解与精度控制上的不足,通过建立一套规范的命名规则与处理流程,将用户的局部干预高效转化为全局的、可自动执行的建模指令,从而实现从复杂网格到高质量CSG实体的稳健重建.本文的主要贡献如下.

(1) 提出了一套完整的半自动交互式逆向CSG建模流程.该流程整合了交互式网格分割、基于命名规则的批量基元拟合及自动化布尔运算,平衡了自动化需求与复杂模型重建精度之间的矛盾.

(2) 设计了一种面向布尔运算的模型命名与数据管理机制.通过制定一套专门的OBJ文件命名规则与中间文本格式,显式编码了基元的类型、布尔运算顺序与层级关系,确保了从网格分割到最终实体生成全过程的信息传递与自动化.

(3) 改进并优化了关键几何基元的拟合方法.针对圆柱、圆锥等基本体,改进了其参数的提取与表示方式,以生成更适于布尔运算、具备完整性与连续性的实体几何模型,从而提升了整体重建的稳健性与成功率.

§2 相关工作

2.1 CAD模型重建

由3D网格或点云构造实体几何属于逆向工程CAD重建问题,可以实现原始数据转换为可供工业编辑和使用的实体模型^[3].重建过程主要包含网格/点云分割^[9-10]、分割单元拟合^[11-13]、各单元实体生成^[14]及实体间布尔运算^[15].

深度学习可以解决原始模型的分割^[16-17]、识别几何图形与基元类型^[18-20],依赖于ABC^[21]、Fusion 360^[22]、SketchGraphs^[23]等大型CAD数据集,而这些方法缺乏构建实体模型所需的CAD建模操作^[3].构建不同复杂度的CAD模型数据集,可实现网格的实体重建^[24-25],采用递归神经网络可以构造实体几何的神经形状分析器^[26-27],把形状解析为一系列CSG操作,通

过基元检测将复杂形体视为一组布尔运算的组合. 基于深度学习的构建方法, 通常更适用于处理拉伸体. 深度学习方法在自动化程度上具有吸引力, 但其“黑箱”特性与对数据集的强依赖性, 使其在处理超出训练集分布的、具有特殊工程要求的复杂模型时能力不足, 重建结果的可控性与精度是其主要瓶颈.

2.2 网格模型分割

网格分割是采取特定的分割准则将较为复杂的网格形体分割成一定数目、符合简单基本形状特征且各自连通的操作^[28-30]. 网格分割方法从用户角度可分为自动分割与交互式分割两种.

自动分割方法便于用户使用, 能满足大多数简单、精度要求不高的应用场合. Cohen-Steiner等^[11]首次将形状近似为一个变分几何划分问题, 通过迭代优化将3D网格自动划分为若干小平面对. Wu等^[31]扩展并优化了变分形状近似技术, 除了最初方法仅适用于平面外, 还涵盖了球体、圆柱体和滚球混合面片, 用于更好地逼近输入网格模型. 二次曲面模型可用于描述不同形状的3D目标, 在网格分割上也得到了应用^[32-33].

交互式分割相对自动分割在实施上较为繁琐, 但能够弥补现阶段自动分割在语义分割上的不足. Meng等^[34]将交互式分割方法分为5类: (1) 基于区域增长算法的交互分割^[35], 其特点是速度快. (2) 基于随机游动分割方法^[36], 对网格中的噪声具有鲁棒性. (3) 自下而上聚合分割算法^[37], 不需要显式的连通性信息. (4) 基于图切的分割^[38], 可以在复杂的图形上指定约束. (5) 交互式寻找切割轮廓以从给定网格中提取组件^[39], 能较好地反映人体形状感知结果. 基于二次曲面拟合的交互式半自动分割方法, 使得分割的效果更为准确、真实, 便于应用在机械模型等分割后需要拟合的较高精度场合^[13, 30]. Wei等^[40]进行网格模型的近似凸分解, 通过全局搜索以平面对模型进行分割, 得到不同的组件及细节, 但针对不同模型的分解阈值需要独立设置, 存在一个面被分解为多个面的情况.

尽管自动分割方法效率高, 但其分割边界往往与工程语义边界不符, 难以准确分离出用于后续布尔运算的纯净基元. 交互式分割虽能弥补此缺陷, 但现有研究多聚焦于分割算法本身. 缺乏将其与后续的CSG重建流程进行系统性整合的框架, 导致从分割到建模的过渡仍依赖一定的手动操作.

2.3 基元拟合

基元拟合在本文中是网格分割的后续操作, 是对每个基元网格近似求解而获得基本参数的过程. 场景中仅含有一类基元模型可通过基于法线的区域增长^[41]或聚集聚类^[42]进行平面检测, Hough变换^[43]或均值移位聚类^[44]可进行球体检测. 以边界感知的几何方法可以实现多种基元模型的拟合^[45], 局限性在于难以提取出独立基元. Schnabel等^[12]基于RASAC^[46]范式提出的点云基元自动检测算法具有快速和稳健等特性, Matt^[47]在MathWorks社区发布的3D二次曲面(椭球、球体、平面、圆柱体、圆锥体等)工具集具备高精度和高鲁棒性, 其中运用了非线性规划求解器减小拟合误差.

现有基元拟合算法在孤立点云上表现稳健, 但在处理从复杂网格中分割出的面片时, 其拟合结果常存在偏差, 直接用于实体构造会引发布尔运算的容错问题. 因此, 需要针对CSG建模的需求, 对拟合结果进行二次优化与校准.

§3 本文方法

本文工作以网格为对象, 向实体转换需要网格模型分割、分割单元拟合及布尔运算等主要部分.

3.1 网格模型分割

本文以类天宫飞行器的两舱体网格模型为研究对象, 如图1(a)所示, 以下简称两舱体, 其中包含常见模型的各类分割问题, 对于处理类似模型同样有效.

采用文献[13]中的方法对模型进行分割, 结果如图1(b), 将整器分割为不同的基元, 含有圆锥、圆柱、长方体及平面等类型.

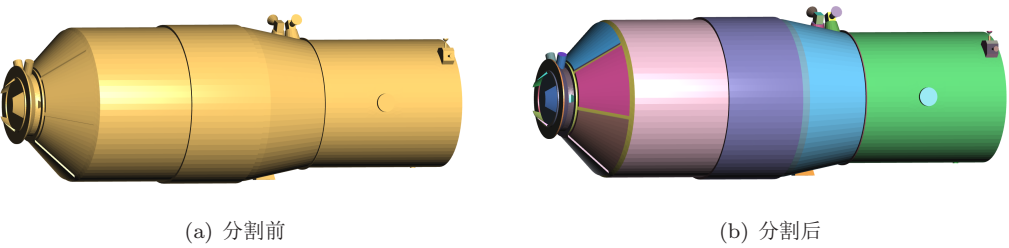


图 1 两舱体网格模型

图1b中获得了各类独立的面, 可以单独表示基元, 而原始模型中的一个基元含有多个面. 如1个圆柱体由1个圆柱面和2个圆平面组成, 而拟合中尽可能减少数据运算量及干扰, 只需其中的圆柱面作为基元拟合数据, 而长方体至少需要2个面.

以用户交互的方式, 点选出相应基元的面, 将分割结果形成独立文件以OBJ格式导出, 获得的四种典型分割模型如图2所示.

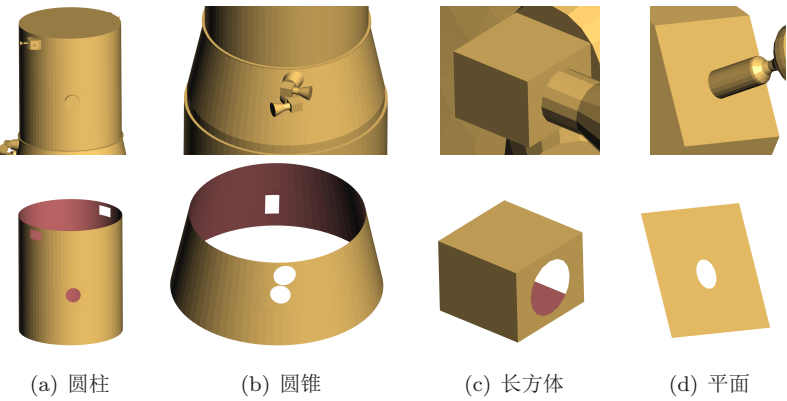


图 2 四种典型分割模型

3.2 分割单元拟合

两舱体分割后获得的子部件仍是面网格结构, 需采取相应算法对其进行拟合, 获取基本体参数.

3.2.1 圆柱体特征

根据文献[12]中的拟合算法, 可以获得圆柱体几何中心 P_h 、轴向矢量 $\mathbf{v}_0$ 、圆柱半径 r 及半高 h_f 这4个基本参数, 如图3(a)所示. 这4个参数可直接构造半个圆柱体, 若想获得整个圆柱体, 则需进行镜像操作, 或采用轴向相反建立另一半圆柱体再布尔“并”成整个圆柱体. 这样的方法虽然能形成圆柱体, 但是增加了运算量, 而且获得的圆柱面是上下两个面, 较难保证几何连续.

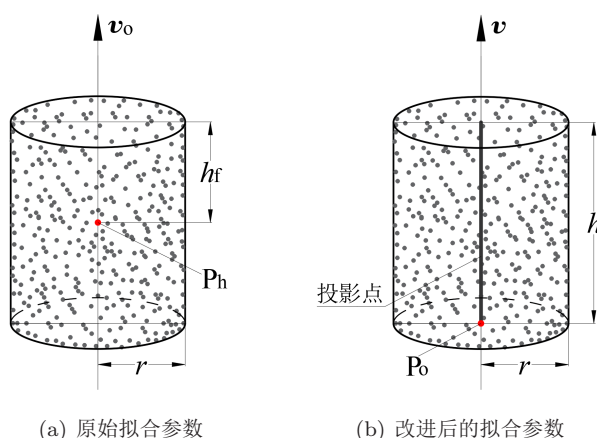


图 3 圆柱拟合

为降低计算复杂度与解决几何连续问题, 本文对该部分进行改进, 获取可直接使用的一组参数, 变动部分如图3(b)所示的基准点 P_o 、圆柱体朝向 $\mathbf{v}$ 及整高 h . 通过以下步骤实现.

步骤1. 以原有方法计算出的几何中心及轴向构建1条空间直线 L .

步骤2. 将原始网格模型的点集 P 向 L 进行投影获得投影点集 P_L .

步骤3. 在 P_L 中, 寻找最大距离的两点. 该过程需要遍历所有的点, 前2个点记录为 P_{m1} , P_{m2} , 得到它们的距离 d_m . 继续取 $P_i (i = 3, 4, \dots, n)$.

步骤3.1. 计算出 P_i 到 P_{m1} , P_{m2} 的距离分别为 d_1, d_2 .

步骤3.2. 比较 d_m, d_1, d_2 的大小, 最大值对应的点替换原有的 P_{m1}, P_{m2} 及 d_m .

步骤3.3. 返回到步骤3.1, 直至遍历完最后1个点. 则获得整高 $h = d_m$, 圆柱底部中心坐标 $P_o = P_{m1}$, 新的轴向 $\mathbf{v} = \overrightarrow{P_{m1}P_{m2}}$.

则圆柱体新的4个基本参数为 $P_o, \mathbf{v}, r, h$. 以图2(a)为例, 拟合出的圆柱体基本参数见表1, 其中点位的绝对误差为拟合值与实际值的距离, 向量的绝对误差为夹角大小, 后续表中同理.

3.2.2 圆锥体特征

根据文献[47]中的拟合方法, 可以获得圆锥体的顶点坐标 P_o 、锥度 θ 、最大圆锥高 $h_{\max}$ 、最小圆锥高 $h_{\min}$ 、俯仰角 pitch , 偏航角 yaw 和滚转角 roll 等参数. 这些参数具备表示出整个圆锥体, 为

表 1 拟合出的圆柱体参数

参 数	拟 合 值	实 际 值	绝 对 误 差
P_0	(0.0029, -0.0031, -0.0012)	(0.0000, -0.0003, -0.0012)	0.0040
$\boldsymbol{v}$	(-1.0000, 0.0000, 0.0000)	(-1.0000, 0.0000, 0.0000)	0.0000
r	1387.4991	1387.5000	0.0009
h	3200.0059	3200.0000	0.0059

表 2 拟合出的圆锥体参数

参 数	拟 合 值	实 际 值	绝 对 误 差
P_0	(3556.5132, 0.0000, 0.0000)	(3556.5710, 0.0000, 0.0000)	0.0578
$\boldsymbol{v}$	(-1.0000, 0.0000, 0.0000)	(-1.0000, 0.0000, 0.0000)	0.0000
θ	0.2025	0.2025	0.0000
$h_{\max}$	8156.5138	8156.6056	0.0918
$h_{\min}$	6776.5128	6776.6056	0.0928

了便于后续实体的生成, 需获取方向向量 $\boldsymbol{v}(v_x, v_y, v_z)$, 做了如下转换

$$\begin{cases} v_x = \cos(\text{pitch}) \cos(\text{yaw}), \\ v_y = \cos(\text{pitch}) \sin(\text{yaw}), \\ v_z = -\sin(\text{pitch}). \end{cases} \quad (1)$$

则圆锥体最终的基本参数为 P_0 、 $\boldsymbol{v}$ 、 θ 、 $h_{\max}$ 及 $h_{\min}$. 如图4所示. 以图2(b)为例, 拟合出的圆柱体基本参数见表2.

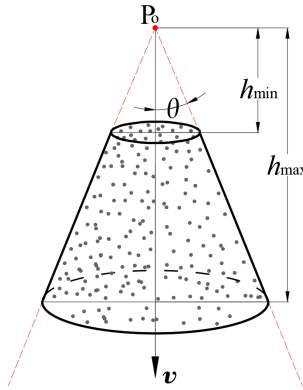


图 4 圆锥拟合参数

3.2.3 长方体特征

对于长方体特征采用包围盒(oriented bounding box, OBB)算法^[48]进行求解, 在开源库Open CASCADE(OCC)^[49]中, 可采用OBB函数实现最小包围盒计算. 计算得到的参数有中心坐标 $P_h(p_x, p_y, p_z)$ 、方向向量 $\boldsymbol{v}(v_x, v_y, v_z)$ 、半长 l_h 、半宽 w_h 、半高 h_h 、局部坐标向

表 3 拟合出的长方体参数

参 数	拟 合 值	实 际 值	绝对误差
$\mathbf{v}$	(0.0000, 0.9397, -0.3420)	(0.0000, 0.9399, -0.3415)	0.0005
$\mathbf{x}$	(-0.0000, -0.3420, -0.9397)	(0.0000, -0.3420, -0.9397)	0.0000
P_o	(-3362.9999, 1628.0710, 69.7748)	(-3363.0000, 1628.0700, 69.7720)	0.0030
l	204.0029	203.9990	0.0039
w	160.0001	160.0000	0.0001
h	204.0074	204.0025	0.0049

量 $\mathbf{x}(x_x, x_y, x_z)$ 、 $\mathbf{y}(y_x, y_y, y_z)$ 、 $\mathbf{z}(z_x, z_y, z_z)$ 。由于参数较多,可进一步简化,以便存储及读取。

将坐标由中心位置变换到其中一个角点,同时坐标系位置也由中心平移到该处,即可构造出一个与包围盒相同的长方体。本文采取了对中心坐标进行变换

$$P_o^T = \mathbf{v}^T - \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \end{pmatrix}^T \begin{pmatrix} l_h \\ w_h \\ h_h \end{pmatrix}. \quad (2)$$

其中 P_o 为角点坐标。

根据坐标系参数的转换规则,同时采用 $\mathbf{v}$ 、 $\mathbf{x}$ 即可确定角点坐标系朝向,新的长 l 、宽 w 、及高 h 为 l_h 、 w_h 、 h_h 的2倍。则长方体最终的基本参数为 $\mathbf{v}$ 、 $\mathbf{x}$ 、 P_o 、 l 、 w 、及 h 。以图2(c)为例,拟合出的长方体基本参数见表3。长方体特征拟合方法在本文中也应用于非规则方体结构,需要配合平面特征进行切割获得最终形体。

3.2.4 平面特征

本文中的平面特征主要运用在楔块状非规则方体的拟合,在获得最小包围盒之后,用平面对其进行切割,获得最后的形状特征。采用文献[12]中的拟合算法,输入平面网格点云,输出平面参数(a 、 b 、 c)、中心位置 P_o 及最大半长 $d_{h\max}$ 。其中平面参数构成以下的(3)过原点的平面方程; P_o 用于约束平面的空间位置; $d_{h\max}$ 为点云包围盒长、宽、高中最大值的一半,用于后续平面的尺寸大小控制。

$$ax + by + cz = 0. \quad (3)$$

以图2(d)为例,拟合出的平面参数见表4,其中 P_o 的绝对误差较大,是由于拟合值为平面离散点的均值坐标,实际值为有界平面的几何中心,但两者均处于同一平面,不影响最终平面的精确表示。

3.2.5 命名规则及中间文件生成

在基本体拟合方法实现之后,需要对每个网格的信息进行依次存储,因此制定了OBJ文件命名规则、设计了专门的中间文本格式,用于后续连续运算的文件直接调用。

根据OBJ文件命名规则自动对所有模型进行拟合,将参数依次输出到文本文件,每个模型有特定的标识符,包含模型顺序、模型类型以及布尔关系,数据段则为此模型的基本参数。如

表 4 拟合出的平面参数

参 数	拟 合 值	实 际 值	绝对误差
a	0.0000	0.0000	0.0000
b	1.0000	1.0000	0.0000
c	0.0000	0.0000	0.0000
P_o	(-215.9922, 1193.2404, 1013.7977)	(-230, 1193.24, 1016.4650)	14.2595
d_{hmax}	130.0000	130.0000	0.0000

图5所示, 标识符“1_0BOXA”中的“1”为布尔顺序, “BOX”表示类型为长方体. 类型后的字母表示操作方法, 如“A”表示主体模型(可作为被修改对象), “T”则为工具模型(用于修改主体模型). 数据段含有数据说明及对应的具体数值.

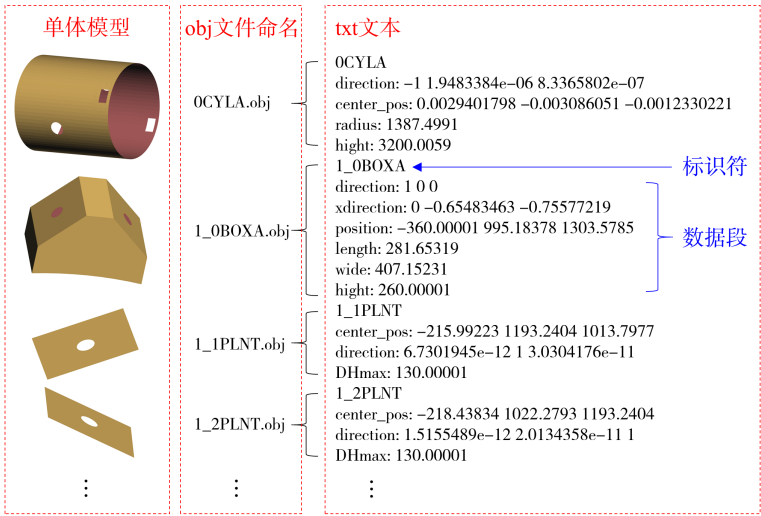


图 5 单体模型批量生成参数文本

3.3 布尔运算

基于OCC库实现实体生成及布尔运算, 文本文件中的基本体参数是根据OCC的函数输入格式进行定制. 由于不是所有部件都由圆柱体、长方体及圆锥体直接“并”获得, 还含有如楔块、凹型体等形状是由“交”、“减”等操作获得. 因此, 无需修改的形体定义为一 级布尔运算, 而需要通过除“并”操作外的其他操作, 定义为多级. 布尔运算步骤如下.

步骤1. 读取第*i*个模型, *i*从第1个模型取到最后1个, 判断它是否需要进行二级运算, 若是则与其工具模型进入步骤1.1, 否则存为临时文件SolidL并进入到步骤2.

步骤1.1. 多个模型作多级运算, 其结果存为SolidL.

步骤2. 判断SolidL是否为第1个模型, 若是, 则令SolidF=SolidL, 否则SolidL与SolidF“并”运算获得新的SolidF.

步骤3. 判断SolidL是否为最后1个模型, 若是, 则结束运算, 最终布尔结果SolidF, 否则 $i := i + 1$, 并返回步骤1.

3.3.1 独立实体生成

基于OCC中BRepBuilderAPI_MakeShape类的函数可将文档中每一个标识符对应参数生成实体, 两舱体各部件网格所生成的基本体实体模型如图6所示.

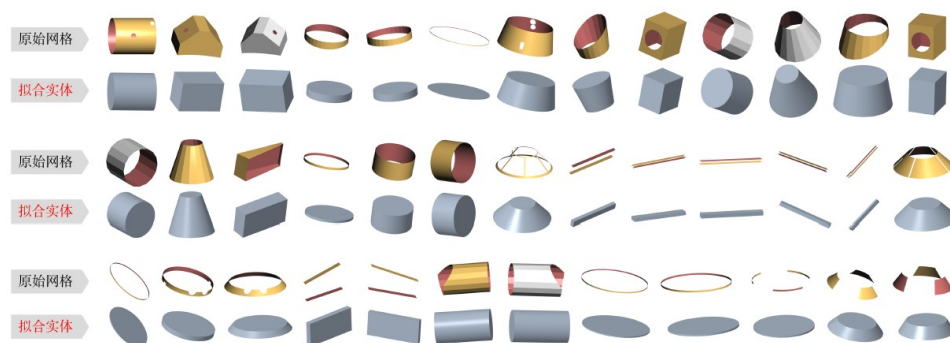


图6 两舱体网格模型对应生成实体

现需要将图6中的所有模型布尔运算成一个实体, 而OCC当前难以将若干模型同时并在一起. 如果同时并运算, 虽不会出现报错现象, 但仅获得装配体模型, 各个实体几何之间依然独立, 仅外表形似整体. 因此, 需要确定布尔连接顺序, 如树的生长过程, 先有主体枝干, 再生长到分支树梢等位置.

3.3.2 一级运算

一级运算是模型主干间的“并”运算, 需基于顺序连接关系. 如图7所示圆点部分模型, 作为主干部分, 布尔操作由简到繁, 对此模型是从右向左依次“并”运算, 主干模型上可以紧接着布尔分支模型, 逐渐连通的关系.

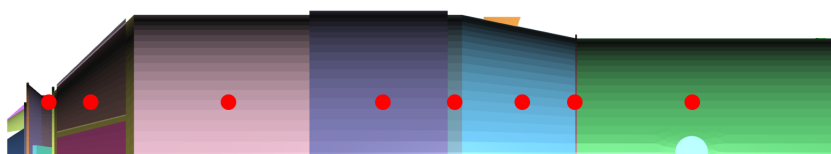


图7 主干模型

如图8所示, 6CONA是作为主干模型, 已经“并”运算到该位置后, 进一步可以“并”上模型11CONA, 再依次12BOXA、13CYLA、14CONA.

拟合运算属于近似问题求解, 拟合出的模型与原始模型存在微量误差, 如位置及方向上存在偏移, 这些会造成布尔失败, 如类似V型夹缝的不完全接触情形. 模型检测到的细节问题, 在误差

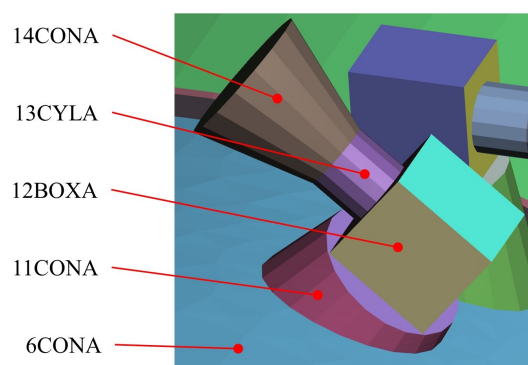


图 8 分支上的一级运算

许可范围内设置较大的布尔运算容差,可明显降低布尔运算的失败率.如果容差设置较大仍不能正确布尔运算,则存在较大的拟合误差,需对拟合算法进行优化.

3.3.3 多级运算

多级运算涵盖了二级、三级甚至更高级,等级越高,其树结构越复杂.如图9所示,二级运算对主体模型修剪时,棱体15.0BOXA需要先进行包围盒运算获得长方体,再采用原始网格所在平面将多余部分切除.该棱体与主干模型接触的面会在后续的“并”运算中自动处理,另外的五个面需要在此作为修剪工具.二级运算出现情形较多,以“T”为工具标识,“-”为树结构划分,如15.1PLNT-15.5PLNT用于分割15.0BOXA.如果是三级运算,则有2个“-”标识,如5.2.0CYLA表示第5组的第2分支的第0个基本体,更多级以此类推,本文中的模型尚未涉及三级及以上运算.

多个平面对模型15.0BOXA的切割如图10所示,每个面可将模型切成两部分,保留体积最大的部分,作为下一个面的切割对象,依次运算剩下的工具面. OCC中的BOPAlgo_MakeVolume类可以分别添加长方体及各个面至同一个容器中,一次性计算获得所有的分割体, BRepGProp类可以查找其中体积最大的体.

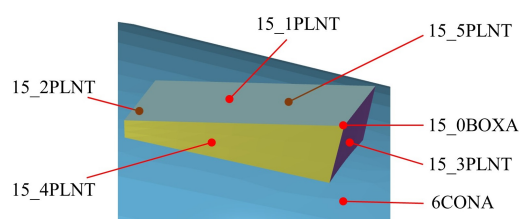


图 9 楔块状棱体

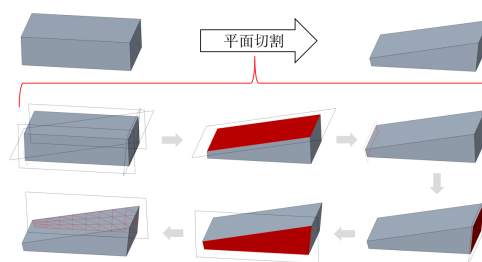


图 10 模型15.0BOXA上的二级运算

二级运算中除了面的切割之外,还存在体的切割,通过布尔“减”实现.如图11的20_0CONA,含有5个如同瓦片形式的扇形组成,它们大小不同,但同属于一个锥体.该锥体等同于被5个长方体分割开,由于获取夹角存在误差且工作量较大,难以采用阵列方式获得其形体,需对5个位置长方体分别进行拟合,再实施布尔“减”运算.将长方体放置到圆锥相应位置,如图12(b)所示.

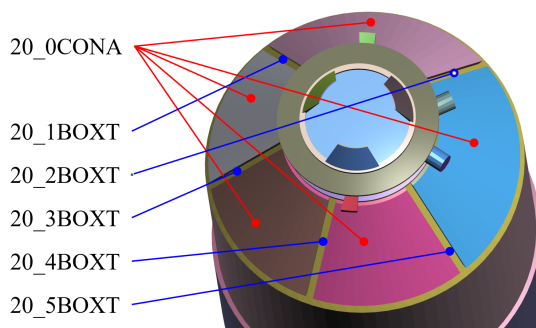


图 11 模型20_0CONA示意

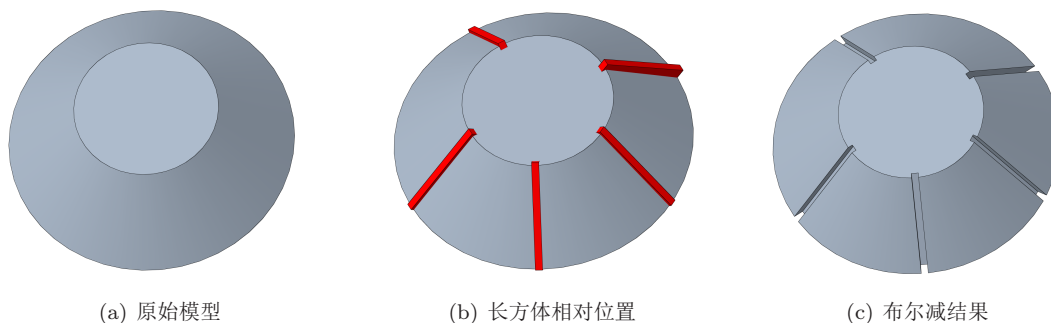


图 12 模型20_0CONA布尔减运算

图12中含有5次布尔“减”运算,通过主体模型20CONA依次减去各个长方体,结果如图12(c)所示,此时可与其他主干模型进行一级运算.

§4 实验结果与分析

本文实例模型为两舱体模型与穿孔模型,前者的实体生成是验证本文方法的有效性,后者用于体现本文方法的特点.

4.1 两舱体模型

两舱体网格模型的逆向CSG建模结果如图13所示,主干部分的实体化均已实现,其剖面如图13(b)所示,内部显示实心剖面,表明整体模型已具备水密性.

两舱体模型属于复杂构型,本文提出的方法暂未实现网格到实体的完美生成,存在的不足如图14所示的资源舱尾部位置,棱体的两个侧面连接了自由旋转曲面,所涉及到的样条拟合算法

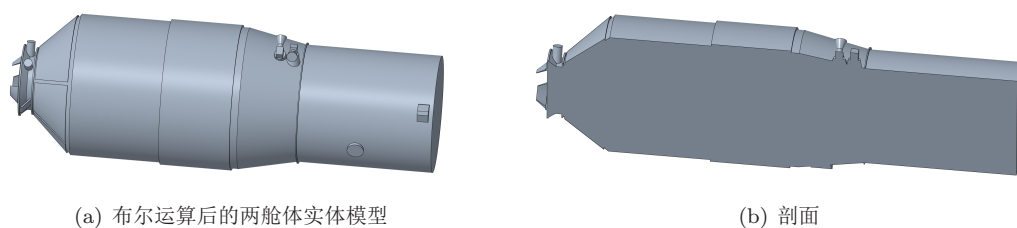


图 13 两舱体模型的实体化结果

或旋转算法在本文研究中暂未形成应用,以至于此处的实体几何生成受阻.可采取的解决思路如下.

步骤1. 将轴对称结构的所有点向对称轴所在任一平面进行投影,获得平面点云.

步骤2. 采用边界提取方法获得轮廓点.

步骤3. 以轮廓点为型值点进行样条曲线拟合.

步骤4. 通过OCC的旋转函数获得封闭壳体.

图14中除了自由曲面难以拟合之外,还存在网格分割缺陷,分割边缘没达到平滑效果,呈现出锯齿状,会导致该位置的上下两部分在拟合及衔接时出错.

另一个不足如图15所示,原始网格中含有斜插构型,是由两个同心圆锥布尔“减”的结果与内圆柱面相交形成.本文方法在网格分割时获得的外圆锥高度与内圆锥高度不一致,实施布尔“减”运算时仅能保留外圆锥原本高度,导致图15(b)无法形成如同图15(a)中的V字构型.

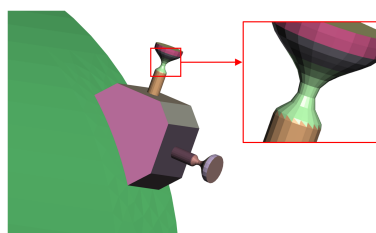


图 14 自由曲面所在位置

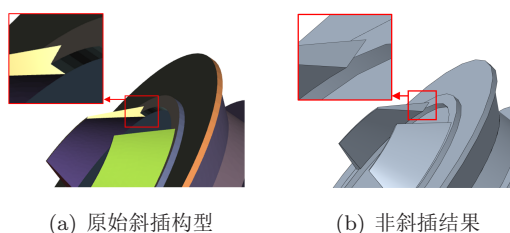


图 15 模型20.0CONA布尔减运算

4.2 穿孔模型

穿孔模型如图16(a)所示,是文献[5]给出的一个难以正确构造CSG模型的问题.该模型主要难点在于扇形镂空区无法采用一次性的平面切割或基本体布尔“减”获得.以本文方法对该模型进行分析,网格分割如图16(b)所示.明确实体构建思路:将外围最大的圆筒与内部平板分开,前者由2个圆柱体布尔“减”获得,后者由1个大圆盘“减”去4个扇形块及2个小圆柱获得,其中的大圆盘由大圆柱让2个平面切割获得.扇形块处理时,先由2个圆盘相“减”,获得的圆柱环再采用扇形块的2个侧平面进行分割,保留小的部分,同样再获得另外3个扇形块.运算关系如图17所示,中下为通过多级处理后的结果.

根据实体构建思路,将各基本面导出并命名,如图17最右栏为各个基元的命名数据,对这些

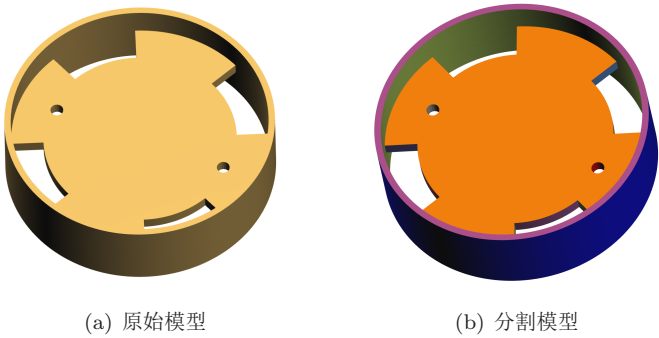


图 16 穿孔网格模型

网格文件进行基本体拟合, 获得中间文本文件, 再经过OCC实体生成及布尔运算获得图17靠右下所示实体结果.

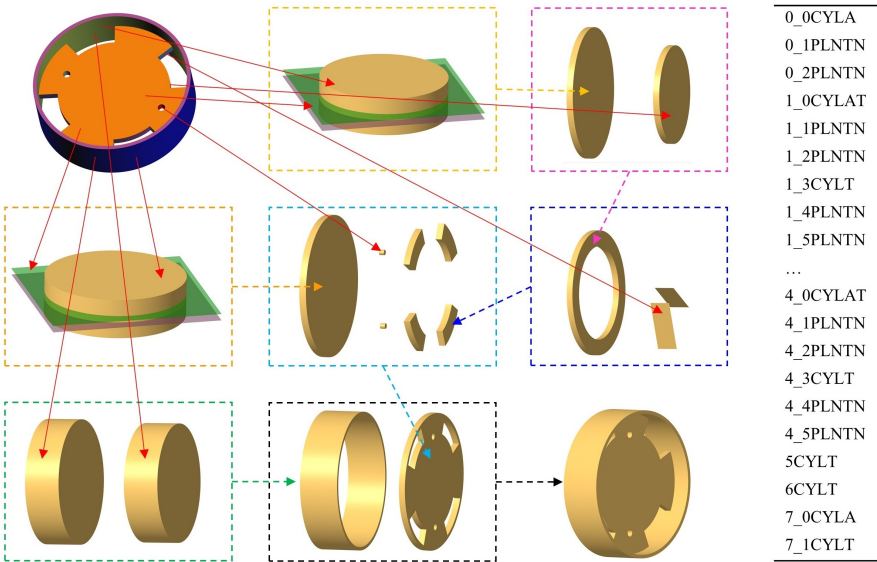


图 17 穿孔模型的基元布尔关系

§5 结语

本文提出了一种半自动交互式CAD网格模型的CSG重建方法, 解决了自动重建方法难以处理的复杂机械模型问题. 以类天宫飞行器的两舱体模型作为研究对象, 制定了布尔运算的模型命名规则, 采用交互式分割方法实现了3D网格的分割, 获得若干个规则命名且具备基本体特征的子网格模型. 改进了基本体拟合方法并对网格模型进行批量处理, 获得具备实体几何模型生成

信息的中间文本文件. 构建了基本体几何生成与布尔运算系统, 实现了中间文件向CSG模型的全自动转换, 成功获得两舱体网格模型的单连通实体几何. 作为方法对比, 在穿孔模型上实现了自动重建方法难以解决的重建工作. 本文方法不足之处在于布尔过程的多级运算灵活性有待提高, 网格分割方法需进一步优化. 未来的研究工作将围绕几点展开: 1) 将自由曲面的重建纳入本框架, 例如通过点云投影与样条拟合来自动生成旋转体; 2) 研究智能分割辅助技术, 利用轻量级交互来引导生成更符合工程语义的分割边界; 3) 优化布尔运算的自动化策略, 研究基于图神经网络的布尔树序列自动生成方法, 以期进一步减少用户干预, 提升方法的智能化水平.

参考文献:

- [1] Li Zhihui, Zhang Hanxin. Gas-kinetic numerical studies of three-dimensional complex flows on spacecraft re-entry[J]. *Journal of Computational Physics*, 2009, 228(4): 1116-1138.
- [2] Li Zhihui, Peng Aoping, Zhang Hanxin, et al. Rarefied gas flow simulations using high-order gas-kinetic unified algorithms for Boltzmann model equations[J]. *Progress in Aerospace Sciences*, 2015, 74: 81-113.
- [3] Li Pu, Guo Jianwei, Zhang Xiaopeng, et al. SECAD-Net: self-supervised CAD reconstruction by learning sketch-extrude operations[J]. *arXiv preprint arXiv*, 2303. 10613, 2023.
- [4] Langbein F C, Marshall A D, Martin R R. Choosing consistent constraints for beautification of reverse engineered geometric models[J]. *Computer-Aided Design*, 2004, 36(3): 261-278.
- [5] Wu Qiaoyun, Xu Kai, Wang Jun. Constructing 3D CSG models from 3D raw point clouds[J]. *Computer Graphics Forum*, 2018, 37(5): 221-232.
- [6] Rabbani T, Van Den Heuvel F. Methods for fitting CSG models to point clouds and their comparison[C]//*Computer Graphics and Imaging: Seventh IASTED International Conference Proceedings*. Kauai: IASTED, 2004, 1719: 279-284.
- [7] Ye Xiuzi, Liu Hongzheng, Chen Lei, et al. Reverse innovative design—an integrated product design methodology[J]. *Computer-aided design*, 2008, 40(7): 812-827.
- [8] Ren Daxun, Zheng Jianmin, Cai Jianfei, et al. CSG-stump: A learning friendly CSG-like representation for interpretable shape parsing[C]//*Proceedings of the IEEE/CVF International Conference on Computer Vision*. Montreal: IEEE Open Journal of the Computer Society, 2021: 12478-12487.
- [9] Shen Liyong, Wang Mengxing, Ma Hongyu, et al. A framework from point clouds to workpieces[J]. *Visual Computing for Industry, Biomedicine, and Art*, 2022, 5(1): 21.
- [10] Zhang Long, Guo Jianwei, Xiao Jun, et al. Blending surface segmentation and editing for 3D models[J]. *IEEE Transactions on Visualization and Computer Graphics*, 2020, 28(8): 2879-2894.
- [11] Cohen-Steiner D, Alliez P, Desbrun M. Variational shape approximation[J]. *ACM Transactions on Graphics*, 2004, 23(3): 905-914.
- [12] Schnabel R, Wahl R, Klein R. Efficient RANSAC for point-cloud shape detection[C]//*Computer graphics forum*. Oxford, UK: Blackwell Publishing Ltd, 2007, 26(2): 214-226.
- [13] Yan Dongming, Wang Wenping, Liu Yang, et al. Variational mesh segmentation via quadric surface fitting[J]. *Computer-Aided Design*, 2012, 44(11): 1072-1082.

- [14] Benkő P, Martin R R, Várady T. Algorithms for reverse engineering boundary representation models[J]. Computer-Aided Design, 2001, 33(11): 839-851.
- [15] Requicha A A G, Voelcker H B. Boolean operations in solid modeling: Boundary evaluation and merging algorithms[J]. Proceedings of the IEEE, 1985, 73(1): 30-44.
- [16] Yan Siming, Yang Zhenpei, Ma Chongyang, et al. Hpnet: Deep primitive segmentation using hybrid representations[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. Montreal: IEEE Open Journal of the Computer Society, 2021: 2753-2762.
- [17] Qiao Yiling, Gao Lin, Yang Jie, et al. LaplacianNet: Learning on 3D meshes with Laplacian encoding and pooling[J]. arXiv preprint arXiv, 1910.14063, 2019.
- [18] Guo Haoxiang, Liu Shilin, Pan Hao, et al. ComplexGen: CAD reconstruction by B-rep chain complex generation[J]. ACM Transactions on Graphics (TOG), 2022, 41(4): 1-18.
- [19] Sharma G, Liu Difan, Maji S, et al. Parsenet: A parametric surface fitting network for 3d point clouds[C]//Computer Vision - ECCV 2020: 16th European Conference, Glasgow: Springer International Publishing, 2020: 261-276.
- [20] Wang Xiaogang, Xu Yuelang, Xu Kai, et al. Pie-net: Parametric inference of point cloud edges[J]. Advances in Neural Information Processing Systems, 2020, 33: 20167-20178.
- [21] Koch S, Matveev A, Jiang Zhongshi, et al. ABC: a big cad model dataset for geometric deep learning[C]//Computer Vision and Pattern Recognition. Long Beach: IEEE Computer Society, 2019: 9601 - 9611.
- [22] Willis K D D, Pu Yewen, Luo Jieliang, et al. Fusion 360 gallery: A dataset and environment for programmatic cad construction from human design sequences[J]. ACM Transactions on Graphics (TOG), 2021, 40(4): 1-24.
- [23] Seff A, Ovadia Y, Zhou Wenda, et al. Sketchgraphs: A large-scale dataset for modeling relational geometry in computer-aided design[J]. arXiv preprint arXiv, 2007.08506, 2020.
- [24] Du Tao, Inala J P, Pu Yewen, et al. Inversecsg: Automatic conversion of 3d models to csg trees[J]. ACM Transactions on Graphics (TOG), 2018, 37(6): 1-16.
- [25] Wu Rundi, Xiao Chang, Zheng Changxi. Deepcad: A deep generative network for computer-aided design models[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 6772-6782.
- [26] Sharma G, Goyal R, Liu Difan, et al. CSGNet: Neural shape parser for constructive solid geometry[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE Computer Society Press, 2018: 5515-5523.
- [27] Yu Fenggen, Chen Qimin, Tanveer M, et al. D2CSG: Unsupervised learning of compact CSG trees with dual complements and dropouts[J]. arXiv, 2301.11497v2, 2023.
- [28] Shamir A. A survey on mesh segmentation techniques[C]//Computer graphics forum. Oxford: Blackwell Publishing Ltd, 2008, 27(6): 1539-1556.
- [29] 孙晓鹏, 李华. 三维网格模型的分割及应用技术综述[J]. 计算机辅助设计与图形学学报, 2005, 17(8): 1647-1655
- [30] 吴仲琦, 郭建伟, 肖俊, 等. 基于二次曲面拟合的机械模型交互式分割方法[J]. 计算机辅助设计与图形学学报, 2019, 31(7): 1210-1220
- [31] Wu Jianhua, Kobbelt L. Structure recovery via hybrid variational surface approximation[J]. Computer Graphics Forum, 2005, 24(3): 277-284.

- [32] Yan Dongming, Wang Wenping, Liu Yang, et al. Variational mesh segmentation via quadric surface fitting[J]. *Computer-Aided Design*, 2012, 44(11): 1072-1082.
- [33] Quan Weize, Guo Jianwei, Zhang Xiaopeng, et al. Improved quadric surfaces recognition from scanned mechanical models[J]. *Computer Aided Drafting, Design and Manufacturing*, 2016, 26(4): 9-19.
- [34] Meng Min, Fan Lubin, Liu Ligang. A comparative evaluation of foreground/background sketch-based mesh segmentation algorithms[J]. *Computers & Graphics*, 2011, 35(3): 650-660.
- [35] Ji Zhongping, Liu Ligang, Chen Zhonggui, et al. Easy mesh cutting[J]. *Computer Graphics Forum*, 2006, 25(3): 283-291.
- [36] Lai Yukun, Hu Shimin, Martin R R, et al. Fast mesh segmentation using random walks[C]//*Proceedings of the 2008 ACM symposium on Solid and physical modeling*. New York: Association for Computing Machinery, 2008: 183-191.
- [37] Xiao Chunxia, Fu Hongbo, Tai C L. Hierarchical aggregation for efficient shape extraction[J]. *The Visual Computer*, 2009, 25: 267-278.
- [38] Brown S, Morse B, Barrett W. Interactive part selection for mesh and point models using hierarchical graph-cut partitioning[C] // *Proceedings of the Graphics Interface*. Toronto: Canadian Human-Computer Communications Society, 2009: 23-30.
- [39] Zhang Juyong, Wu Chunlin, Cai Jianfei, et al. Mesh snapping: Robust inter-active mesh cutting using fast geodesic curvature flow[C]//*Computer Graphics Forum*. Oxford: Blackwell Publishing Ltd, 2010, 29(2): 517-526.
- [40] Wei Xinyue, Liu Minghua, Ling Zhan, et al. Approximate convex decomposition for 3d meshes with collision-aware concavity and tree search[J]. *ACM Transactions on Graphics (TOG)*, 2022, 41(4): 1-18.
- [41] Holz D, Holzer S, Rusu R B, et al. Real-time plane segmentation using RGB-D cameras[C] // *RoboCup 2011: Robot Soccer World Cup XV 15*. Berlin: Springer Berlin Heidelberg, 2012: 306-317.
- [42] Feng Chen, Taguchi Y, Kamat V R. Fast plane extraction in organized point clouds using agglomerative hierarchical clustering[C] // *2014 IEEE International Conference on Robotics and Automation (ICRA)*. Hong Kong: IEEE, 2014: 6218-6225.
- [43] Abuzaina A, Nixon M S, Carter J N. Sphere detection in kinect point clouds via the 3d hough transform[C] // *Computer Analysis of Images and Patterns: 15th International Conference, CAIP 2013*. Berlin: Springer, 2013: 290-297.
- [44] Tran T T, Cao V T, Laurendeau D. eSphere: extracting spheres from unorganized point clouds: how to extract multiple spheres accurately and simultaneously[J]. *The Visual Computer*, 2016, 32: 1205-1222.
- [45] Li Duanshun, Feng Chen. Primitive fitting using deep geometric segmentation[C]//*2019 Proceedings of the 36th International Symposium on Automation and Robotics in Construction (ISARC 2019)*. Banff: IAARC Publications, 2019, 36: 780-787.
- [46] Fischler M A, Bolles R C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography[J]. *Communications of the ACM*, 1981, 24(6): 381-395.

- [47] Matt J. Object-oriented tools for fitting conics and quadrics [EB/OL].(2022-10-25)[2023-5-10].<https://www.mathworks.com/matlabcentral/fileexchange/87584-object-oriented-tools-for-fitting-conics-and-quadrics>.
- [48] Gottschalk S A. Collision queries using oriented bounding boxes[D]. Chapel Hill: The University of North Carolina at Chapel Hill, 2000.
- [49] Open CASCADE SAS. Open CASCADE Technology (Version 7.7.0) [EB/OL].(2022-11-07)[2023-03-21]. <https://dev.opencascade.org/release/open-cascade-technology-770>.

Reverse CSG modeling of CAD mesh model

YANG Quan-ou¹, LI Zhi-hui², JIA Xiao-hong³, SHI Wei-bo⁴, YAN Dong-ming⁵

- (1. School of Aerospace Engineering, Guizhou Institute of Technology, Guiyang 550003, China;
- 2. Beijing Aerohydrodynamic Frontier Research Center, Beijing 100011, China;
- 3. Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China;
- 4. Hypervelocity Aerodynamics Institute, China Aerodynamics Research and Development Center, Mianyang 621000, China;
- 5. Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China)

Abstract: In order to solve the problem that the 3D mesh of complex mechanical configuration is difficult to reconstruct the CSG (Constructive Solid Geometry) model automatically, a semi-automatic interactive method for reverse CSG modeling of CAD (Computer Aided Design) mesh model is proposed. The naming rules of Boolean operation model are established. Firstly, a semi-automatic segmentation method with moderate user interaction is used to segment the mesh model. In this process, the segmented model is exported and named according to the Boolean tree structure, and a batch of mesh files with primitive characteristics are obtained. Then, the primitive fitting function is improved to automatically fit the batch mesh files according to the naming relationship, and the intermediate text file for CSG generation is obtained. Finally, the primitive fitting and Boolean operation model with the intermediate text as the input is constructed based on OCC (Open CASCADE), and the automatic generation of CSG model is realized. The CSG re-construction is carried out on the complex mesh model of Tiangong-type spacecraft, and most of the details of the model are obtained, which has highly consistent surface characteristics with the original model. The method is applied to the pierced pattern model which is difficult to construct by automatic method, and the corresponding solid model is successfully obtained.

Keywords: mesh segmentation; constructing CSG; naming rules; primitive fitting; boolean operation

MR Subject Classification: 68U05