



山东大学
SHANDONG UNIVERSITY

毕业论文(设计)

论文(设计)题目:

复杂几何结构减材制造的连续路径规划

姓 名 _____ 史子涵
学 号 _____ 201900130126
学 院 _____ 计算机科学与技术学院
专 业 _____ 计算机科学与技术
年 级 _____ 2019 级
指导教师 _____ 赵海森

2023 年 05 月 23 日

摘 要

目前主流的两制造方式分别是增材制造和减材制造。增材制造即常听到的 3D 打印，减材制造则是使用 CNC 数控机床将坯料切削成需要的形状。减材制造常使用到的数控机床主要为三轴数控机床，四轴数控机床和五轴数控机床。其中，使用三轴数控机床进行复杂零件的加工时，需要人工进行材料的多次放置和组装。如果转为使用四轴数控机床增加自由度并进行路径规划，则可以大大减少人工的参与。增材制造被认为是快速成形技术的一种，但在实际应用中，减材制造仍是目前工业最广泛使用的制造方式，拥有增材制造所没有的优点，如加工速度快等。基于此背景，本研究面向四轴数控机床减材制造的连续路径规划。

本文中的算法为已有工作内容，该算法面向四轴数控加工规划连续路径，增加了四轴数控加工领域中路径规划算法的选择。该算法目的在于通过规划四轴机床上对坯料进行减材制造时刀具的连续路径，避免人工在加工过程中的参与，从而将坯料加工成特定形状。本人工作为在碰撞检测部分添加刀具向量组，以避免实际应用中的碰撞。四轴数控机床，也就是 3+1 数控机床应用广泛，联动型的路径规划问题具备普遍性，本项目有很强的实际应用潜力。

关键词：CNC 加工；减材制造；四轴数控机床；连续路径规划

ABSTRACT

Currently, the two main manufacturing methods are additive manufacturing and subtractive manufacturing. Additive manufacturing is generally known as 3D printing, while subtractive manufacturing mostly uses CNC machines to cut stock into the desired shape. The CNC machine tools commonly used in subtractive material manufacturing are mainly three-axis CNC machine tools, four-axis CNC machine tools and five-axis CNC machine tools. When using three-axis CNC machine tools to process complex parts, it is necessary to manually place and assemble materials several times. If the four-axis CNC machine tool is used to increase the degree of freedom and carry out path planning, the manual participation can be greatly reduced. Additive manufacturing is considered as one of the rapid prototyping technologies, but in practice, subtractive manufacturing is still the most widely used manufacturing method in industry, which has advantages that additive manufacturing dose not have, such as fast processing speed. Based on this background, it is necessary to research the continuous path planning for material subtractive manufacturing of four-axis CNC machine tools.

The algorithm in this paper is existing job, it is orient to the continuous path planning of four-axis CNC machining, which increases the choice of path planning algorithm in the field of four-axis CNC machining. The purpose of this algorithm is to process the stock into a specific shape by planning the continuous path of the cutting tool when the stock is reduced on the four-axis CNC machine tool, avoiding the manual participation in the process. My work is to add cutter vector group to the collision detection part to avoid collision in practical application. Four-axis CNC machine tools, that is, 3+1 CNC machine tools, are widely used, and the problem of linkage path planning is universal. This project has strong practical application potential.

Keywords: CNC machining, Subtractive manufacturing, Four-axis CNC machine tool, Continuous path planning

目 录

第 1 章 绪论	1
1.1 选题背景和研究意义	1
1.2 文献综述	3
1.3 论文组织结构	4
第 2 章 连续路径规划算法	5
2.1 输入数据及处理	5
2.2 碰撞检测	9
2.3 全局连接	11
2.3.1 生成独立路径段	11
2.3.2 独立路径段连接为一条路径	12
2.3.3 方向均匀化	13
2.4 算法改进	13
2.5 改进结果展示	16
第 3 章 刀具添加算法表现	18
3.1 测试数据	18
3.2 测试结果	18
第 4 章 总结与展望	22
4.1 本文总结	22
4.2 工作展望	22
参考文献	23
致 谢	25

第1章 绪论

1.1 选题背景和研究意义

目前的机械加工方法可以分为增材制造，等材制造和减材制造三种基本方式。增材制造有广为人知的 3D 打印技术，拥有设计自由，大规模定制，废料最少化以及制造复杂结构的能力以及快速原型制作的优势，因而在过去几十年中发展迅速^[1]。等材制造中的注塑成型作为应用最广泛的制造方式，占据了很大的市场空间。但是在大部分情况下，被广泛使用的减材制造拥有更加明显的优势。首先，在面对复杂几何结构时，增材制造大多数时候需要人工进行加工的设计，进行多次放置和拼接；比如 Cignoni P 等人在论文^[2]中提出的技术，就需要将三维数字形状分解为一组易于制造和组装的互锁部件。而减材制造是不同类型的数控加工和材料去除工艺的总成，这些更常见地由计算机数控(CNC)驱动。这样的加工方式保证了加工成品的完整性，使得成品相较于增材制造更加坚固，其中的五轴技术更是具有高精度、高效率、高柔性等特点^[3]。除此之外，在工业生产中，减材制造相比于增材制造效率更高。

减材制造中，数控加工是主流加工方式。数控加工是指，由控制系统发出指令使刀具作符合要求的各种运动，以数字和字母形式表示工件的形状和尺寸等技术要求和加工工艺要求进行的加工^[4]。它泛指在数控机床上进行零件加工的工艺过程。

本文中所使用到的四轴数控机床主要用于四轴数控铣加工，用来加工直径和重量比较大，或在卧式车床上难于安装的工件，以及径向尺寸大而轴向尺寸相对较小，形状复杂的大型和重型工件。本文算法面向加工复杂几何形状或结构的加工目标，适合选用四轴数控机床进行加工。不仅如此，四轴数控机床既拥有比三轴数控机床更高的自由度，同时使用维护等花费又不会像五轴数控机床那样高。

四轴数控机床也被称为 3+1 数控机床。3 代表四轴数控机床有三个基础的直线轴（移动轴），1 代表四轴数控机床有一个旋转轴。四轴数控机床的旋转轴为绕 X 轴的旋转轴，如图 1-1 所示，为四轴数控机床的三个直线轴 X, Y, Z 和一个旋转轴 A。图 1-1 中，带有孔洞的圆盘部分用于安装工件，安装在其上的长方柱形

体为待加工的工件，上方的细长圆柱形体为刀具。可以看到，安装工件的圆盘可以带着工件沿 X 、 Y 、 Z 轴任意方向移动，这样可以让刀具在工件当前面上进行任意位置的加工；安装工件的圆盘还可以带着工件绕 X 轴，也就是以旋转轴 A 轴的方向旋转，以能够让刀具加工到工件的每一面。

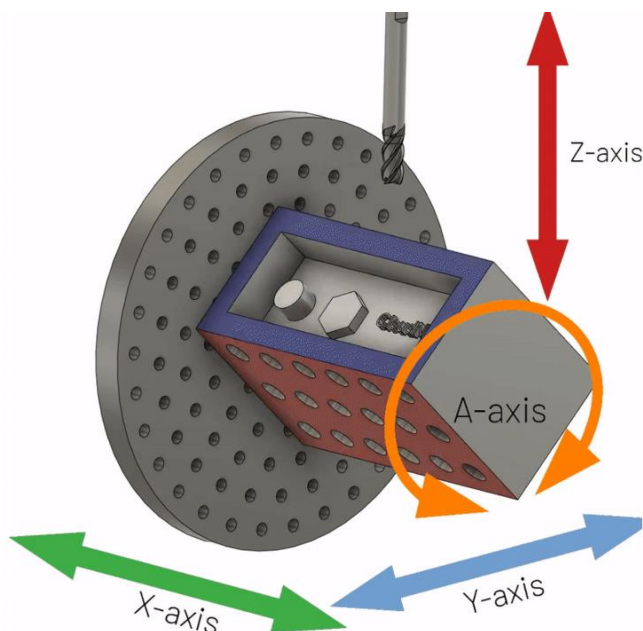


图 1-1 四轴数控机床的四条轴^[5]

四轴数控机床这一领域包括许多方向。在减材制造的机床加工领域，智能机床的研究与应用是未来的主要发展方向，包括研究智能机床的关键技术，复杂型面件数字化加工的工艺流程，曲面件数字化加工刀具轨迹规划等问题^[6]。本文算法就是面向复杂几何结构的连续路径规划，隶属于曲面件数字化加工刀具轨迹规划这一问题。该算法可以生成基于四轴数控机床加工方式的刀具的连续加工路径，目的在于避免人工在加工过程中的参与，来将坯料加工成特定的形状。目前商用软件自动生成的路径在路径连续性上有待改进，本文算法有希望很大程度上提升 3+1 基础的加工表面质量，并同时降低加工时间。目前国内外有关路径规划的算法有很多，这些将在 1.2 节中进行综述。本文内容包括两部分，一部分是已有工作内容，另一部分是在已有工作内容上解决刀具的碰撞检测问题。其中，已有工作内容的主要难点在于全局路径的连接，使用到一些算法如图割算法等来解决，这一部分将在 2.3 节内提到。四轴数控机床应用广泛，联动型的路径规划问题具有普遍性，算法将拥有很强的实际应用潜力。

1.2 文献综述

3+1 数控机床广泛应用于加工各种复杂结构工艺品和复杂结构工业零件，对于数控机床上面向复杂几何结构的连续路径规划是未来主要研究方向中的一个重要话题，拥有很强的实际应用潜力。本小节将主要论述在这一方向上，国内外就连续路径规划的研究现状。

在寻找有关四轴数控加工减材制造相关的文献时，发现了一些增材制造领域中的路径规划算法。如 Lensgraf S 等人在论文^[7]中提出一种新的 3D 工具路径生成范式用于熔丝制造成型技术(FFF)，使其无需将对象的局部特征限制在单层打印。由于 FFF 技术是从无到有的制造方式，在制造过程中与 CNC 加工有很大的不同。这一文献中的连续路径规划算法并不适用于 CNC 加工。

前面提到了 CNC 加工的一些限制，其中一个重要问题就是数控机床上刀具与坯料之间避免发生碰撞的问题。实际上，目前国内外已经提出了许多 CNC 加工中碰撞检测的算法。如 Suh S H 等人在论文^[8]中提出了四轴加工中刀具干涉的算法，包括两种三轴加工中同样存在的干涉和一种四轴加工中特有的干涉。本文算法使用到了这篇论文中将刀具体积分解为一个半球和一个圆柱体的思想：首先将刀尖半球简化为一个刀尖半径长度的向量以模拟刀具的刀尖，再将刀具的剩余部分抽象成向量组。在搜寻文献时，也发现了很多基于五轴数控加工的碰撞检测算法，虽然没有能够应用在本文算法中，但是可以获得对碰撞检测更深的了解。如 Youssef A 等人在论文^[9]中提出了五轴数控机床加工中避免刀具发生碰撞的算法，以及 Tang, Duc T 等人在论文^[10]中综述了五轴数控加工中碰撞检测与避免算法的研究现状和比较。

在刀具路径的规划方面，Abidin 等人在论文^[11]中对刀具路径策略进行了比较，并选择具有最大体积去除量的最短刀具路径，对 B 样条模型的加工工艺进行了优化，以显著提高生产率并降低生产时间。Lee J N 等人在论文^[12]中提出了一种在五轴机床上加工的刀具轨迹生成方法，同时可以通过齐次坐标变换，生成四轴加工的侧铣刀具位置。由于应用场景不同，这两种方法实际上没有使用到本文中。

最后是 CNC 加工流水线的设计方面的文献。本文参考了论文^[13]，在这篇文章中，使用到了将坯料与旋转轴正交分割的思想，即将坯料正交于旋转轴进行切片，进而研究刀具在每一层切片上的连续路径规划问题。

1.3 论文组织结构

本文一共分为四章。

第 1 章：绪论。简要介绍了面向复杂几何结构的连续路径规划的选题背景和研究意义，在文献综述部分对国内外这一方面的研究进行了总结，并在最后介绍了本文的组织结构。

第 2 章：算法。本章的主要内容包括按照算法执行顺序介绍算法，以及完成碰撞检测部分未考虑到的刀具的碰撞检测问题。其中，2.1-2.3 为算法介绍，2.4-2.5 为碰撞检测问题的解决。

第 3 章：结果。在完成第 2 章提到的内容后，选取了一些输入数据进行测试，并将输入数据和输出结果列在本章中。

第 4 章：讨论。对本文进行总结，并且列出本文相关方向的工作展望。

第2章 连续路径规划算法

本文工作基于已有工作，完成路径优化中未考虑的刀具碰撞检测问题。因此本章内容分为两部分，2.1-2.3 节为算法介绍，这部分是已有工作。2.4-2.5 节为碰撞检测问题增添刀具建模向量组和展示。在进行本章内容的详细介绍之前，首先概括本文算法思想如下：

算法思想为“由内向外”或“自底向上”，先分析局部的可用刀具范围，再进行全局连接。

1. 路径采样带点做碰撞检测，计算出每个点的可用刀具范围。这一部分基于方向采样的方法，对于每个采样方向做碰撞检测（第 2.2 节）。
2. 使用 GraphCut 算法生成独立的路径段（第 2.3 节）。
3. 用一个基于最小生成树求解的 TSP 算法，将独立的路径段连接为一条路径（第 2.3 节）。
4. 在可用刀具范围内做 Laplace 平滑，得到均匀化方向（第 2.3 节）。

2.1 输入数据及处理

该算法的输入数据为常见的 obj 文件。如图 2-1 所示，这是一个玉如意模型的正视图。红色标注的部分是机床的移动轴 x 轴方向。首先要将模型进行变换处理，使其位于三维坐标轴上的正确位置，以便于后续对模型进行加工时将该模型与以 x 轴正交的方向对模型进行切片。

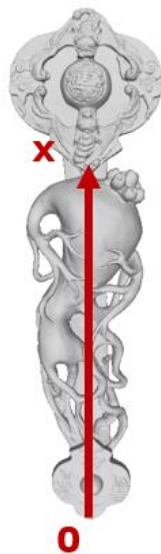


图 2-1 玉如意模型及机床移动轴的 x 轴

将模型移植三维坐标轴正确位置后，将该模型以与 x 轴正交的方向对模型进行切片，得到一组与 x 轴正交的切片。由于玉如意模型是一个复杂结构的几何体，其切片上将会有许多轮廓。如图 2-2 就是玉如意模型沿 x 轴进行切片后，在 $x=20$ （单位为毫米）处的切片截图。图中的每一个圈都是当前切片上截出来的一个轮廓。



图 2-2 在 $x=20$ （单位为毫米）处玉如意的切片

为什么需要将输入的模型数据处理成一层层的切片数据呢？切片主要原因如下所述：该算法主要面向精加工，因此用于加工的坯料已经拥有一定与轮廓大致相同的外形，在加工过程中，刀具旋转围绕这些轮廓对坯料进行加工。本文算法要做的事情就是避免刀具与轮廓发生碰撞以及规划刀具的连续路径。如图 2-3 就是刀具围绕这一切片上的轮廓进行加工的刀具路径示意图。该图中，刀具用一条线段表示，刀具在多个点上加工（这也是图中有许多线段的原因），将坯

料加工成需要的形状。

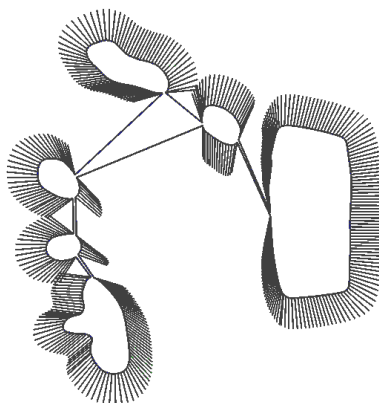


图 2-3 刀具围绕 $x=20$ (mm) 处切片上面的轮廓进行加工

因此，从图 2-3 可以得知，刀具将在每一层切片上有一个二维层次上的路径规划。也就是说，本文算法最开始需要面向切片进行处理。这就是对模型进行切片的原因。对模型进行切片后，获得的切片数据被存入 offsets 文件中。如表 2-1 所示，是玉如意模型的 offsets 文件数据的一部分。

表 2-1 玉如意模型的 offsets 文件数据的一部分

5.0		
10.0		
83		
1252	10	
10	-11.4227	-3.84793
10	-11.4521	-3.85288
.....
1201	15	
15	-11.8433	-3.74299
15	-11.8507	-3.7127
.....
.....

如表格 2-2，是对文件中数据的意义进行说明。

表 2-2 offsets 文件数据意义

数据	数据意义
5.0	每层切片之间的距离。该数据仅在 offset 文件起始处出现一次。

10.0	第一层切片的 x 坐标值。该数据仅在 offsets 文件起始处出现一次。
83	所有切片中轮廓的个数。该数据仅在 offsets 文件起始处出现一次。
1252 10	当前轮廓中点的个数，以及当前轮廓所在切片的 x 坐标值。该数据将在每个轮廓的点组前出现一次。
10 -11.4227 -3.84793	当前轮廓中一个点的三维坐标。多个点组成一个轮廓的点组。

以上全部就是将 obj 文件转换成 offsets 文件的数据处理过程及处理原因。在此总结将 obj 文件转换为 offsets 文件的操作流程表如表 2-3 所示：

表 2-3 将 obj 文件转换为 offsets 文件的操作流程表

1) 将 obj 文件进行变换处理，使其能够以一定的位置放置在三维坐标轴上。本处理过程中需要将 obj 模型沿 x 轴对齐，并且底部能够位于原点处；
2) 将 obj 文件转换为 off 文件，这一步调整数据格式是为了能够调用下一步所需要的函数；
3) 调用 CGAL 的函数，将 off 文件沿 x 轴进行切片，并将切完片的数据另存为 offsets 文件；
4) obj 文件处理完成。

获得切片数据（存储在 offsets 文件中）后，接下来将继续对 offsets 文件中的数据进行处理，以获得用于下一步碰撞检测的数据。

由于 offsets 文件是按照轮廓进行分组的数据，因此从 offsets 读入数据后，首先将读入的数据以轮廓为单位进行存储。其中，每个轮廓中将会有许多个点（以三维点向量的形式存在）。对于这些数据，对其进行的处理如表 2-4 所示：

表 2-4 对 offsets 文件数据的处理流程表

1) 进入循环，对于每一个轮廓：
1.1) 对轮廓中的点进行等间距划分。注意：这一步骤会使点的数量发生变化；
1.2) 更新点集为步骤 1.1) 之后的点集；
1.3) 去除掉过小的轮廓，并将轮廓中点的存储顺序全部修改为顺时针方向。
2) 后处理部分：由于 offsets 文件中的数据是以轮廓为单位进行存储的。在获得步骤 1) 处理后的数据后，对这部分数据进行后处理，将其以切片为单位进行分割，同时，每个切片上会有一个或多个轮廓，每个轮廓上会有多个点。以此形式对数据进行存储，便于后续算法对数据的使用。

需要注意的是，在获得处理好的数据后，接下来进入循环，以每一个切片为

单位执行后续算法。也就是说，2.2 节至 2.3 节算法实际上是面向一个切片层的数据进行处理的。经过 2.1 节所述处理完成的数据以切片层-轮廓-三维点向量的层次存储。

2.2 碰撞检测

由 2.1 节可知，输入数据为一个切片层的数据。在详细说明这些数据的处理之前，首先来介绍碰撞检测。

碰撞检测问题的具体应用十分广泛。例如，虚拟环境应用中的飞行员和宇航员的培养与训练、机器人的路径规划和学习、交互式动画系统、服装 CAD 中衣物与人体躯干的配合等。碰撞检测问题按运行物体所处的空间可分为二维平面碰撞检测和三维空间碰撞检测^[14]。本文算法虽然是面向复杂几何结构进行加工的规划算法，但是本文采用将复杂几何结构进行分层，对每一层进行处理的思想。这样就将一个需要进行三维空间碰撞检测的问题转换为对每一层切片进行碰撞检测的二维平面碰撞检测问题。这也是 2.1 节中对数据进行后处理时，将数据以切片为单位进行存储的原因。

如图 2-4 所示，蓝色的线表示轮廓，红色的箭头表示可加工方向。对一个切片上的每一个轮廓上的每一个点 P，从一个点可以延伸出许多可加工方向，如 A 和 B，这些可加工方向以一个向量的形式进行存储。首先，这些方向向量不可以与点所在的轮廓相交，基于此条件，可以得到每个点的可加工方向角度范围；其次，这些方向向量应该避免和同一切片中的其他轮廓相交，也就需要进行碰撞检测。因此，由第一个条件得到每个点的可加工方向范围后，在该角度范围内进行均匀采样，得到一组可加工方向向量并存储，等待进行碰撞检测。

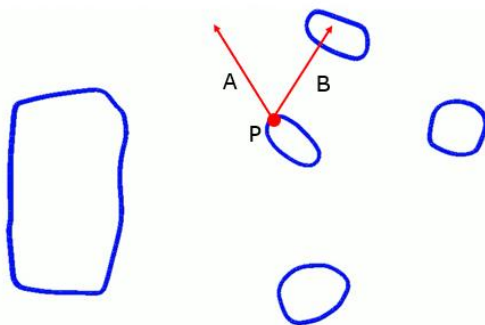


图 2-4 某切片中某一轮廓上某点的两个可加工方向

可以看到，在图 2-5 中，A 和 B 都是点 P 的可加工方向。很明显，如果刀具在 B 方向上对点 P 进行加工，则刀具会与其他轮廓发生碰撞；如果刀具在 A 方向上对点 P 进行加工，则不会发生碰撞。碰撞检测就是检测在某一可加工方向上，刀具是否会与其他轮廓发生碰撞。进行碰撞检测后，将只留下不会发生碰撞的可加工方向，参与到后续的算法中。

碰撞检测算法实现基本流程表如表 2-5 所示。需要注意，碰撞检测算法的输入是模型的一个切片数据。

表 2-5 碰撞检测算法基本流程

对于当前切片中的每一个轮廓 c:

1) 对于当前轮廓 c 中的每一个点 v:

1.1) 计算每个点 v 在轮廓 c 上的切向向量 **tangent**;

1.2) 计算每个点 v 在轮廓 c 上的法向向量 **normal**;

1.3) 计算每个点 v 的可加工方向范围（以角度的形式进行保存。本算法中使用角度作为输入参数进行向量的旋转）。这一步骤的目的在于通过限定可加工方向范围，排除点 v 的某一可加工方向 **angle** 会与点 v 所在的轮廓 c 发生碰撞的可能。

2) 已经在 1) 部分获得了进行碰撞检测所需要使用到的基础向量 **tangent** 和 **normal**。1.1) 部分的循环结束后，开启新的循环，对于当前轮廓 c 中的每一个点 v:

2.1) 从点 v 的可加工方向范围中，每隔一定角度进行采样得到一个可加工方向 **angle**，并将点 v 的所有 **angle** 进行保存，等待进行检测;

2.2) 对于该点 v 的每一个可加工方向 **angle**:

2.2.1) 首先将刀具抽象为一条沿点 v 的法向量 **normal** 无限延长（使用足够大的数字作为向量长度，以接近无限的概念）的向量 **m**;

2.2.2) 将该向量 **m** 进行 **angle** 角度的旋转，得到旋转后的向量 **m₁**，用于模拟刀具位于某一可加工方向;

2.2.3) 对于当前切片中的所有轮廓，判断旋转后的向量 **m₁** 是否会与任何一个轮廓相交，如果会，说明该可加工方向 **angle** 上的刀具将会与当前切片中其他轮廓发生碰撞，舍弃该可加工方向 **angle**。

经过碰撞检测算法后，就获得了当前切片中所有轮廓上所有点的所有可加工方向。刀具在这些可加工方向上对点进行加工时，将不会与当前切片中的任一轮廓

廓发生碰撞。这些可加工方向将用于后续的路径生成和连接。

2.3 全局连接

本小节算法依然面向模型的一层切片上的数据进行处理。本小节所需要使用到的数据包括两个部分，第一部分是 2.1 节对输入的 `offsets` 文件数据进行了处理并按切片进行分割之后的数据，在本小节中将使用到 2.1 节处理后数据中当前切片层上的数据；第二部分是 2.2 节算法面向一个切片层进行碰撞检测后得到的可加工方向数据。由于 2.2 节算法和本小节算法同属一个循环下（对于每一个切片的循环），在本小节中将使用同一次循环下 2.2 节的所有输出数据。

2.3.1 生成独立路径段

首先简要介绍这部分的算法实现思路：对每一个点，选取一个可用的可加工方向，使用 `GraphCut` 算法将轮廓上可以进行连续加工的点进行连接生成独立的路径段。

算法实现思路细化如下：

首先，计算路径段是本文中生成独立路径段算法的第一步。一个路径段意味着一系列沿着轮廓的端到端的点。其中，每个点只有一个可加工方向，在这样一个段中，每两个相邻的点之间的可加工方向有重叠区域，对重叠部分的检测是生成独立路径段算法的一个关键操作。

其次，关于本算法中所使用到的 `GraphCut` 算法。该算法使用了 `Schmidt F R` 等人在论文^[15]中提出的一种用于平面图的快速图形切割算法。该算法使用平面图切割库计算平面图上的最大流量/最小流量，实现了高效的算法，具有几乎线性的运行时间。该库还提供了几个易于使用的接口，以便定义计算机视觉应用中常见的平面图。这一算法基于图论工作，拥有高效的特性。可以将该算法应用于形状匹配和图像分割。

生成独立路径段算法实现基本流程表如表 2-6 所示。

表 2-6 生成独立路径段算法基本流程

- | |
|--|
| <p>1) 对于当前切片中的每一个轮廓：</p> <p>1.1) 进入 <code>while(true)</code> 循环。在每一次循环中，任选一个有效的可加工方向，沿着当前轮廓向前或向后遍历，直到遇到一个无效的可加工方向。在这样的一次循环</p> |
|--|

中，就获得了一个路径段。多次循环，获得多个路径段，直到所有有效可加工方向都已经用于生成轮廓上的路径段。

- 1.2) 使用图割操作获得每个路径段上点的可加工方向。在上一步碰撞检测中，输出的是每个点的许多个可用的可加工方向。在本算法中经过图割，计算出每个点在它所在路径段上的一个可用可加工方向。
- 1.3) 进行后处理。对当前轮廓中的每一个点，对于这个点的多个可加工方向，找到图割计算出的可加工方向，保存并作为整个路径段生成算法的输出。

上述算法完成后，当前切片中所有轮廓上的每一个点就有了唯一可用于生成路径的可行的可加工方向，并且同一路径段上的点将拥有同一标志，用于标记不同的路径段。

于是再次进入新的循环，将当前切片中已经标记好的属于不同路径段的点连接为路径段。将这些点连接为路径段的算法与标记这些点的算法过程类似。如表 2-7 所示，为该算法的过程。

表 2-7 将不同路径段的点连接为路径段

对于当前切片中的每一个轮廓：
1) 进入 while(true)循环，在每一次循环中：
1.1) 从一个拥有执行方向的点开始（该执行方向是指在图割操作中计算出的可加工方向），沿着当前轮廓向前或向后遍历，将执行方向具有重叠部分的点连接为路径段。这样，每次循环可以获得一个路径段。
1.2) 如果路径段过小，则不做处理，默认舍弃；否则将其存储入专门用于存储路径段的数据结构中
2) 多次循环后，获得当前切片中的所有路径段。

全部上述处理完成后，该算法输出计算的得到的多个路径段。获得这些路径段之后，为了保证路径连接的稳定性，将检查全部路径段。一旦有路径段长度过小或者可加工方向太少，就将该路径段去除，不进入后续的路径连接中。完成上述所有实现后，将当前切片中的已经计算好的路径段作为输入，进入到下一部分：独立路径段连接为一条路径。

2.3.2 独立路径段连接为一条路径

首先简要介绍这部分的算法实现思路：该算法使用一个基于最小生成树求解

的 TSP 算法，将 2.3.1 节中生成的一层切片中所有轮廓上独立的路径段连接为一条路径。所使用到的 TSP 的点是路径段的端点，TSP 的边是不同路径段之间端点的连线。本算法是一个非完全图的问题，因为某些两个端点之间的连线可能会是无效的（当连线与其他点发生碰撞时，该连线被认为是无效的）。

将独立路径段连接为一条路径的算法实现基本流程表如表 2-8 所示：

表 2-8 独立路径段连接为一条路径的算法流程

1)	将所有路径段上的点标记为该点所在路径段的序号，记录路径段的端点，准备好作为 TSP 点的数据；
2)	计算这些路径段中每两个端点之间的距离，得出一个距离矩阵，准备好作为 TSP 边的数据；
3)	基于最小生成树对该问题进行求解，返回 TSP 路径上所有边的总权值和 TSP 路径上的点；
4)	记录单路径以及单路径上的可加工方向。

将独立路径段连接为一条路径后，得到当前切片中多个轮廓上的多个路径段连接而成的单路径，并记录了单路径的加工方向。

2.3.3 方向均匀化

在经过 2.3.2 节处理后，获得了当前切片上的单路径。这部分函数通过使用拉普拉斯的迭代过程对单路径进行方向均匀化处理。首先获取初始加工方向，通过拉普拉斯迭代在可用刀具范围内平滑加工方向，直到单路径上连续点的角度变化足够小。该函数执行后，输出均匀化后的加工方向。

方向均匀化的算法实现基本流程表如表 2-9 所示：

表 2-9 方向均匀化的算法流程

1)	记录初始（默认）加工方向；
2)	使用 do...while 循环，通过拉普拉斯迭代，平滑加工方向，直到路径上角度变化足够小。

进行方向均匀化后，得到当前切片上的均匀路径。

2.4 算法改进

由 2.2 节可知，碰撞检测部分并没有考虑到实际加工中刀具的形状，而是使

用了一条方向向量作为刀具进行碰撞检测。因此，此处将讲述对这部分的改进。

首先进行调研，得到市面上常见的刀具的形状如图 2-5 所示：



图 2-5 刀具的形状

本文所设计的刀具形状与上图类似，同时刃尖保持了对称。如图 2-6 所示：

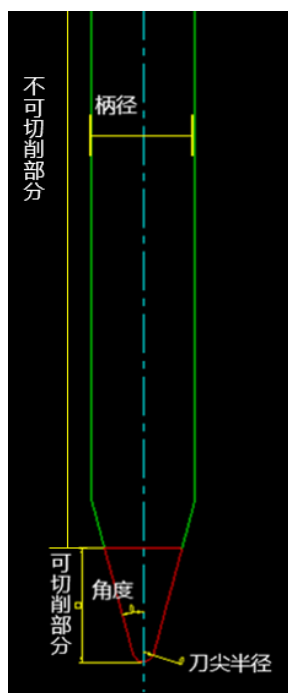



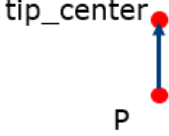

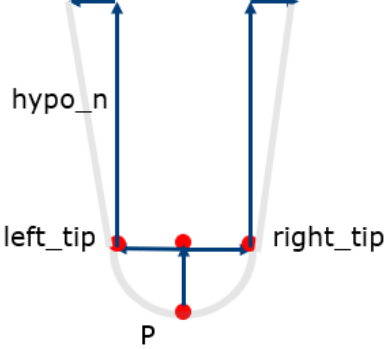
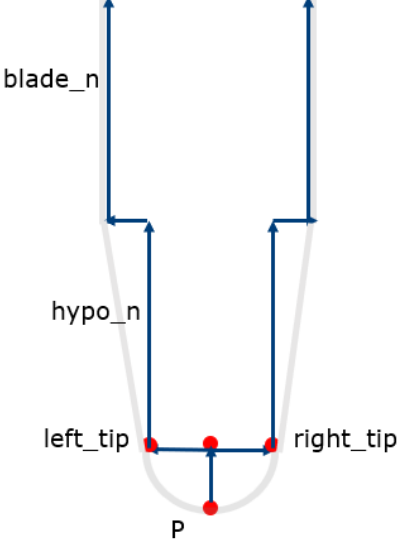
图 2-6 刀具的形状建模

了解刀具的形状及参数意义之后，接下来开始构建刀具的形状向量。

首先可以知道，每个点是存在于轮廓之上的（最初数据处理时已经将少于三个点的轮廓去除）。由第 2.2 节可知，每个点的法向量和切向向量可以由该点与相邻两点计算得出。在拥有了法向量和切向向量的基础上，对刀具的形状进行构

建如表 2-10 所示：

表 2-10 对刀具形状的建模流程

建模细节	示意图
对于轮廓上的每一个点 P：	
该点加上刃尖半径长度的法向量，可以得到刃尖的圆心：	
从刃尖圆心分别减去和加上刃尖半径长度的切向向量，可以得到刃尖直径的两个端点：	
根据刀具参数计算斜边部分法向量方向上的向量 hypo_n。分别在刃尖直径的两端加上这一向量，再分别减去和加上一半柄径长度的切向向量，得到刀刃斜边部分的另外两个端点：	
最后，计算刀刃的直边部分法向量方向上的向量 blade_n，分别在刀刃的两个斜边端点上加上，即可获得刀刃的整个向量组的构造。 注：右图只是意指，实际刀刃的直边部分比右图所示要长。	

需要知道向量组的构建是在某一点的法向量方向上进行构建的，且点的可加工方向以角度 **angle** 的形式保存。在获得法向量方向上的向量组后，需要将该向量组绕 x 轴（本算法的输入文件假设沿 x 轴进行切片），也就是旋转轴 A 方向

（如图 1-1 所示）进行 **angle** 角度的旋转，才能获得某一可加工方向上的刀具向量组。

对每一个点的每一个可加工方向，都将会会有一个该可加工方向上的刀具向量组等待检测。如表 2-11 所示，该表中的伪代码说明了对每个可加工方向上的刀具向量组所进行的操作。

表 2-11 刀具向量组碰撞检测伪代码

Vector group collision detection pseudocode	
1:	for 向量组中的每一条向量 v
2:	{
3:	for 当前切片中的每一个轮廓 c
4:	{
5:	if 存在向量 v 与轮廓 c 相交
6:	将向量 v 标记为不可用;
7:	}
8:	}

以上就是刀具建模并添加碰撞检测的详细过程。刀具建模向量组加入碰撞检测，能够避免实际应用中刀具与模型发生碰撞。

2.5 改进结果展示

2.4 节详细介绍了刀具建模的构建。在完成刀具建模向量组后，测试刀具建模向量组是否正确。已知刀具建模向量组首先是在待加工点的法向方向上建立的，再将其按照某一可加工方向的角度旋转后，进行碰撞检测。因此，在进行碰撞检测之前，需要测试法向方向上的刀具建模向量组是否符合刀具形状，以及向量组旋转后，是否符合刀具形状。将这两步分别得到的刀具向量组进行输出并对比，判断是否相同，如果相同，说明上述两步在计算过程中没有发生错误。如图 2-7 所示，该图由某个点法向方向上的刀具向量组的输出绘制而成，可以看到，在左图中，刀具的向量组（HE-EC-CB-BD-DG）构成刀具形状，这说明刀具向量组在法向方向上建模正确；如图 2-8 所示，该图由该点的某一可加工方向上刀具向量组的输出绘制而成。图 2-8 中刀具形状与图 2-7 中刀具形状相同，这说明刀具向量组在经过旋转后，依然保持原有形状。这一结果证明，刀具建模向量组正确，且在旋转后依然正确，即用于碰撞检测的刀具向量组正确。

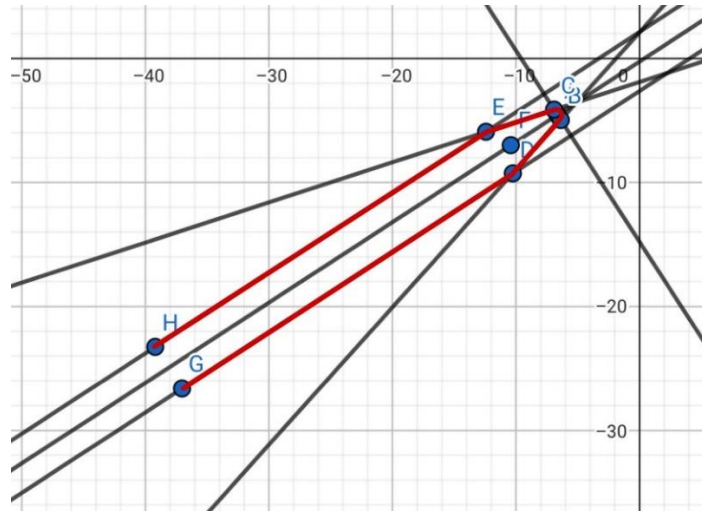


图 2-7 某点法向方向上的刀具向量组绘制图

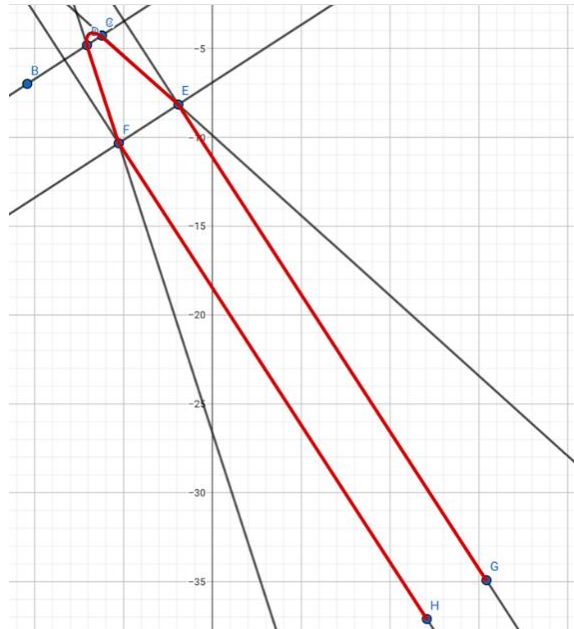


图 2-8 某点某可加工方向上的刀具向量组绘制图

第3章 刀具添加算法表现

本章主要用于展示原有算法的数据生成结果和加入刀具形状测试（2.4 节）后的数据生成结果对比。

3.1 测试数据

测试所使用到的输入数据即为一个玉如意模型的 obj 文件。该模型如图 3-1 所示：

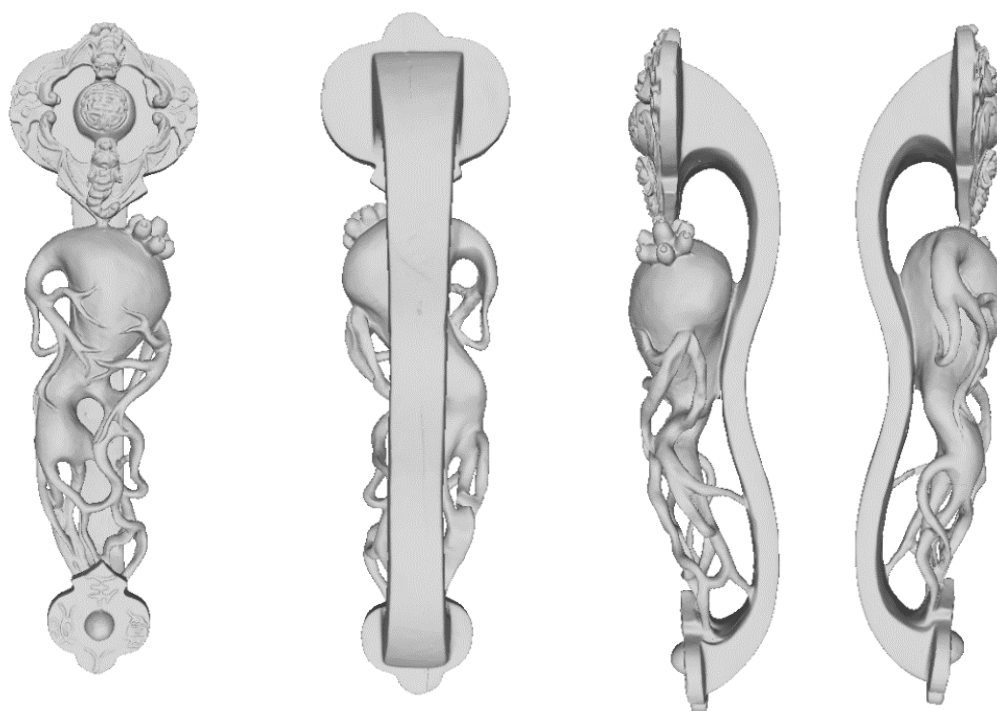


图 3-1 玉如意模型

本文所使用到的输入测试数据模型即为图 3-1 所示模型数据。获得模型数据后，进行第 2 章所述的处理，从而获得输出数据。输出数据的对比在 3.2 节详细说明。

3.2 测试结果

由 3.1 节可知输入数据。在原有算法（将刀具抽象为一条简单的向量进行碰撞检测）中，玉如意模型一共切割出 20 个轮廓数目大于等于 1 的切片。此处列出三个原有算法（将刀具抽象为一条简单的向量进行碰撞检测）和改进算法（将

刀具建模为一组向量组进行碰撞检测)的同一切片上的刀具加工路径对比。以下从图 3-2 到图 3-5, 其中的左图为原有算法输出的刀具加工路径, 右图为添加刀具向量组进行碰撞检测后输出的刀具加工路径。

可以看到, 在图 3-2 的右图中, 部分轮廓因加工时可能会发生刀具碰撞而不进行加工, 刀具线段围成的轮廓不如左图刀具线段围成的轮廓精细。

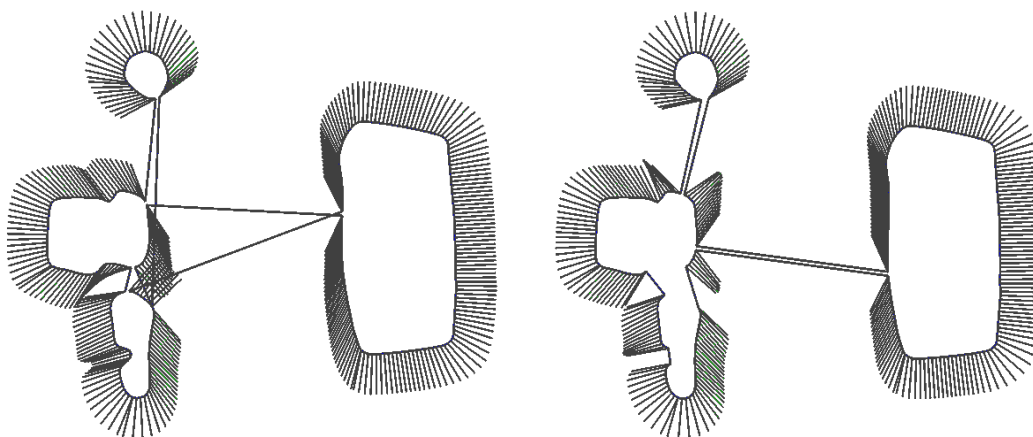


图 3-2 第 2 个轮廓个数不少于 2 的切片上的刀具加工路径

在图 3-3 中, 可以看到距离较为相近的两个轮廓部分由于可能发生碰撞而没有进行加工。

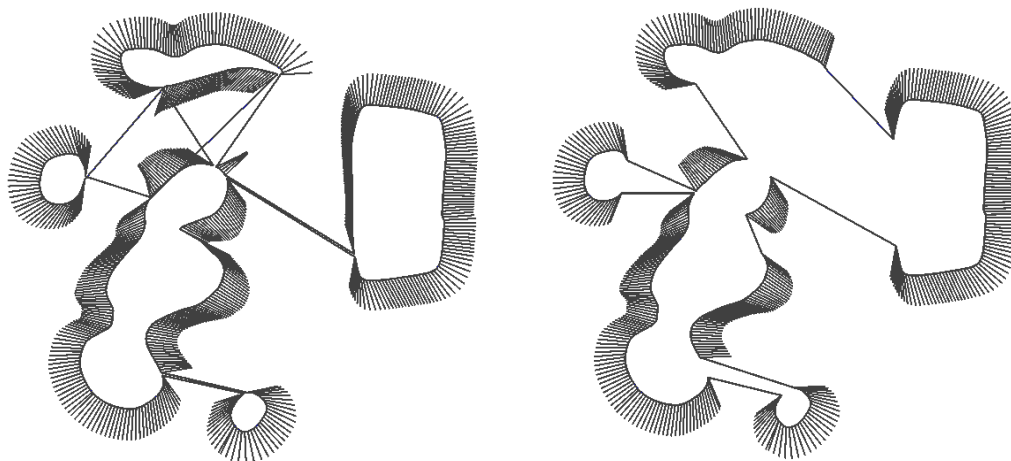


图 3-3 第 7 个轮廓个数不少于 2 的切片的刀具加工路径

在图 3-4 中, 可以发现部分曲率较大的向内凹的轮廓无法被加工。刀具在这部分进行加工时, 极有可能与两边发生碰撞, 导致此种现象。

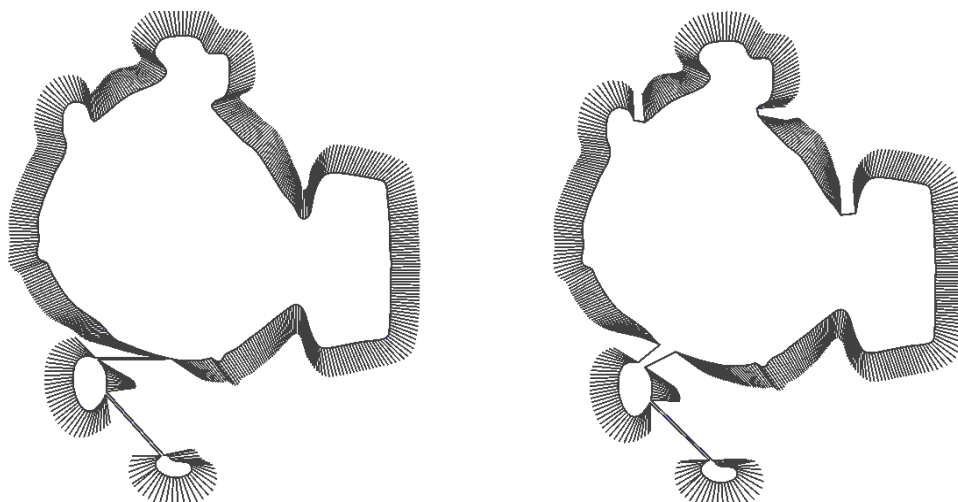


图 3-4 第 13 个轮廓个数不少于 2 的切片的刀具加工路径

在图 3-5 中，可以看到，由于轮廓之间间距足够大，轮廓均可以进行加工，但是部分轮廓的加工方式发生了变化。图中左上方的轮廓，左图和右图的刀具路径不相同。在添加了刀具向量组进行检测之后，避免了与左侧轮廓可能会发生的碰撞，改为从下方对这部分轮廓进行加工。

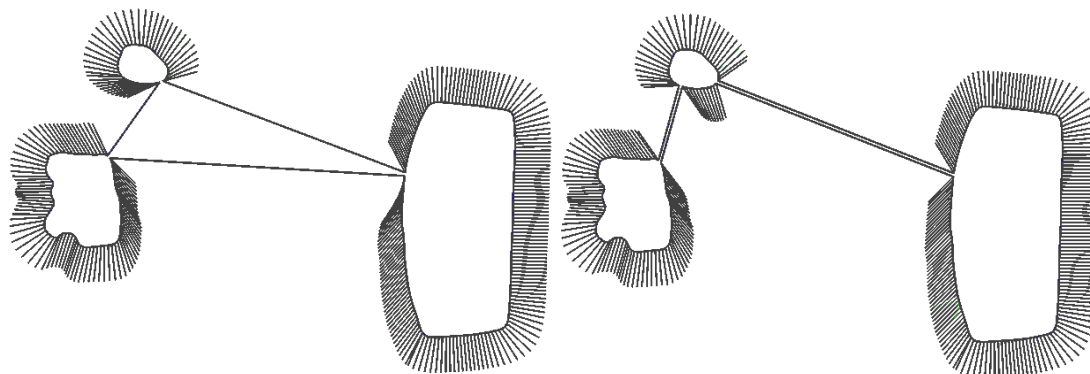


图 3-5 第 18 个轮廓个数不少于 2 的切片的刀具加工路径

观察图 3-2 至图 3-5，可以发现，在加入刀具建模后，刀具加工路径均发生了变化。该项目中建模出的刀具参数如表 3-1 所示，刀具参数意义参考图 2-10 所示，其中，总刀具长度为可切削部分长度和不可切削部分长度的和。

表 3-1 刀具参数表

刀尖半径	0.5mm
刃斜边角度/角度	15°
刀具柄径	4mm
总刀具长度	38mm

综上，可以发现，在添加了刀具向量组进行检测后，避免了实际中可能会发生的刀具碰撞。同时注意到，为了避免发生碰撞，基于该加工方式，舍弃了一部分轮廓的加工，这一点是无法避免的。

第4章 总结与展望

4.1 本文总结

本文提出了面向四轴数控机床的连续路径规划的一种算法。本算法基于四轴数控机床的加工方式，采用将模型进行切片后，对每一层切片进行碰撞检测和路径连接的思想，是一套完整的路径规划算法。

算法提出了减材加工中四轴机床上路径规划的一种方式，四轴机床减材制造工业上应用广泛，且路径规划问题具有普遍性，算法将拥有很强的实际应用潜力。

在学习并添加刀具建模向量组至本算法的过程中，了解了四轴机床减材制造的过程，对国内外这部分的研究内容有了较多的理解，以及对联动型路径规划问题有了更深入的学习。

4.2 工作展望

下步工作的规划主要有两项：

首先，第三章介绍了加入刀具建模前后的刀具路径对比，这部分数据不够充足。期望在下步工作中可以使用更多模型输入数据进行测试。同时注意到，部分轮廓会因为避免刀具发生碰撞而无法被加工，需要考虑何种加工对象更适用于这种加工方式，以减少无法加工的部分，达到精加工的目的。

其次，由 2.4 节可知，刀具建模成的向量组进行碰撞检测时，在每一层切片中，需要检测刀具的每一条向量是否会与当前切片中的所有轮廓发生相交。在这种情况下，算法的时间复杂度会很大，程序执行时间长。为了减少程序执行时间，考虑降低算法复杂度，参考了 Katchalski-Katzir E 等人在论文^[16]中提到的方法，该论文使用了快速傅里叶算法以降低算法的时间复杂度。但是这部分工作并没能显化到本项目中。因此，降低本项目中算法的时间复杂度也是下步工作规划之一。

参考文献

- [1] Chen X , Chen G , Wang G , et al. Recent Progress on 3D cc rinted Polylactic Acid and Its Applications in Bone Repair[J]. Advanced Engineering Materials, 2019, 22(4).
- [2] Cignoni P , Pietroni N , Ponchio F , et al. Interlocking Pieces for Printing Tangible Cultural Heritage Replicas. Eurographics Association, 2014.
- [3] 唐清春. 面向五轴增减材复合成形轮廓精度的运动轨迹控制研究[D]. 湖南大学.
- [4] 熊武一, 周家法, 卓名信等. 《军事大辞海》: 长江出版社, 2000
- [5] [What's the difference Between 3-axis, 4-axis & 5-axis Milling? - CloudNC](#)
- [6] 刘宏发, 姜淑凤, 高福生等. 现代数字化加工技术的问题探讨与增减材制造能力数据分析[J]. 湖南农机, 2020.
- [7] Lensgraf S , Mettu R R . Beyond layers: A 3D-aware toolpath algorithm for fused filament fabrication[C]// IEEE International Conference on Robotics & Automation. IEEE, 2016.
- [8] Suh S H , Lee K S . Avoiding tool interference in four-axis NC machining of rotationally free surfaces[J]. IEEE Transactions on Robotics & Automation, 2002, 8(6):718-729.
- [9] Youssef A . Optimization of machining strategy and process planning of complex geometry.[D]. McMaster University (Canada). 2004.
- [10] Tang, Duc T . Algorithms for collision detection and avoidance for five-axis NC machining: A state of the art review[J]. Computer-Aided Design, 2014, 51:1-17.
- [11] Abidin, Zainal Fahmi Zainol and Muhammed Nafis Osman Zahid. Comparative Study of Tool Path Strategies in CNC Machining for Part with B-spline Surfaces. *Lecture Notes in Mechanical Engineering* (2019): n. pag.
- [12] Lee J N , Huang C B , Chen T C . Toolpath generation method for four-axis NC machining of helical rotor[J]. Journal of Achievements of Materials and Manufacturing Engineering, 2008, 31(2).
- [13] Nuvoli S , Tola A , Muntoni A , et al. Automatic Surface Segmentation for Seamless Fabrication Using 4-axis Milling Machines[J]. John Wiley & Sons, Ltd, 2021(2).
- [14] 王志强, 洪嘉振, 杨辉. 碰撞检测问题研究综述[J]. 软件学报, 1999, 10(5):7.
- [15] Schmidt F R , Toppe E , Cremers D . Efficient planar graph cuts with applications in Computer Vision[C]// IEEE Conference on Computer Vision & Pattern Recognition. IEEE, 2012:351-356.
- [16] Katchalski-Katzir E , Shariv I , Eisenstein M , et al. The Role of Geometric Fit Between Protein Molecules and their Ligands in Determining Biological Specificity 1[J]. Advances in Molecular and Cell Biology, 1996, 15:623-637.
- [17] Young H T , Chuang L C . An Integrated Machining Approach for a Centrifugal Impeller[J]. International Journal of Advanced Manufacturing Technology, 2003, 21(8):556-563.
- [18] 傅伟, 赵东宏, 王庭俊,等. 数控四轴旋转刀轨优化技术的研究[J]. 工具技术, 2017, 51(11):4.
- [19] Liang Y , Zhang D , Chen Z C , et al. Tool orientation optimization and location determination for four-axis plunge milling of open blisks[J]. International Journal of Advanced Manufacturing Technology, 2014, 70(9-12):2249-2261.
- [20] Ren J X , Yao C F , Zhang D H , et al. Research on tool path planning method of four-axis high-efficiency slot plunge milling for open blisk[J]. International Journal of Advanced

- Manufacturing Technology, 2009, 45(1-2):101-109.
- [21] Suh S H , Jih W S , Hong H D , et al. Sculptured surface machining of spiral bevel gears with CNC milling[J]. International Journal of Machine Tools and Manufacture, 2001, 41(6):833-850.
- [22] Yang J , C Araújo, Vining N , et al. DHFSlicer: Double Height-Field Slicing for Milling Fixed-Height Materials[J]. ACM Transactions on Graphics, 2020, 39(6).

致 谢

本科四年的时间被青岛的风一点点吹散，又到了毕业的季节。在此，写下我的致谢，希望能将内心诚挚的谢意传达给重要的人心中。

首先，我非常感谢我的导师赵海森教授。从着手项目，到过程指导，再到论文写作，赵老师给予了许多指导。在他的指导下，我不仅学习到了项目相关的知识，还学到了很多实践技巧和思路。正是赵海森老师的指导，让我能够顺利学习和开展项目，以及完成论文的撰写。

感谢本科四年的任课老师，许多老师认真负责，课堂内容充实，让我能够更加理解和学习课程内容，学习到更多知识。

感谢毕业设计期间其他同样由赵老师指导毕业设计的同学给我提供的帮助，如惠浩同学，乔婧同学和张振民同学。以及赵老师实验组的张志豪同学。同时，我意识到要向有能力的同学学习，要让自己强大起来，并且能够帮助他人。

感谢家人和朋友。在这四年中，感谢家人的支持，让我能够无忧无虑地学习，不必担心生活上的困难；感谢朋友的帮助，在许多个遇到困难的日子里，是季群、程梓芸、张嘉琪、吴澜，马慧颖等朋友的帮助和鼓励让我能够迈过难关。

衷心感谢

