

分形模型的 3D 打印路径规划

翟晓雅, 陈发来*

(中国科学技术大学数学科学学院 合肥 230000)
(chenfl@ustc.edu.cn)

摘要: 为了在保证边缘细节的精度条件下最大限度地提高打印效率, 提出了分形模型的 3D 打印路径规划算法. 将分形模型分为 3 个打印区域: 轮廓层, 过渡层及内层. 针对 3 个不同的区域采取不同的填充措施. 文中算法主要分为以下 4 个步骤: (1) 为了保证边缘细节的精度, 利用轮廓平行算法的高保真性填充分形模型的轮廓层. 将不同等距线进行特征点检测并设置相应的阈值来优化轮廓层层厚; (2) 以过渡层的面积与过渡层内外边界是否相交为条件约束建立优化问题确定过渡层的最优层厚, 并确定一条光滑的三次 B 样条曲线作为过渡层内边界 (即内层边界); (3) 针对过渡层和内层进行全局连续 Fermat 螺线填充, 并提出 Fermat 螺线局部优化的算法; (4) 将 3 个区域内部的填充曲线连接进行全局路径优化. 实验结果表明, 相比于传统的路径规划算法与连续 Fermat 螺线算法, 文中算法具有更高的效率及更好的打印效果, 并且适用于具有复杂边界及多连通区域的几何模型.

关键词: 3D 打印; 分形; 路径规划; 三次 B 样条; Fermat 螺线
中图法分类号: TP391.41 **DOI:** 10.3724/SP.J.1089.2018.16618

3D Printing Path Planning of Fractal Models

Zhai Xiaoya and Chen Falai*

(School of Mathematical Sciences, University of Science and Technology of China, Hefei 230000)

Abstract: This paper presents a path planning algorithm for printing fractal models which guarantees the accuracy of the boundary details and maximizes the efficiency of 3D printing process. We divide the printing region into three layers, namely the contour layer, the transitional layer and the inner layer. According to different layers we use different filling processes. The algorithm is comprised of four main steps: (1) adopt offset path for outer layer to ensure the boundary details, and the thickness of the outer layer is optimized by checking the number of feature points in the offset curves under the given threshold value; (2) determine a fair cubic B-spline curve as the boundary between the translation layer and inner layer, and then optimize the layer thickness of the transitional layer; (3) use a single continuous Fermat spiral to fill in the transitional layer and inner layer, and propose a local optimization algorithm for Fermat spiral; (4) optimize the global path in three regions. We provide several examples to demonstrate the effectiveness and the printing quality of our method. Comparisons are also made with previous approaches which show the superiority of our method. Furthermore, our method works for geometric models that are composed of multiply connected regions with complex boundaries.

Key words: 3D printing; fractal model; tool-path generation; B-spline curve; Fermat curve

3D 打印技术从 20 世纪 90 年代中期出现以来迅速成为制造业领域的一项新兴技术, 并且逐渐应用到了各行各业. 3D 打印技术作为一种高效、分散式的生产工具带我们进入个性化与社会化创造的时代, 并且将商品生产推向多元化与个性化的生产模式.

3D 打印的技术本质是增材制造, 其基本流程为: 首先对 3D 模型进行数据格式转换, 然后进行切片计算和打印路径规划, 最后利用打印机逐层打印出实物. 路径规划是其中的一个重要环节, 它是指 3D 打印模型在从 2D 到 3D 的逐层积累过程中, 根据当前横截面所获得的不同的切片层规划出具体的打印路径并对其进行合理的优化, 得到更快更好的 3D 打印效果^[1-2].

目前 3D 打印路径传统的生成算法主要有 zigzag 路径规划算法^[3]和轮廓平行算法^[4]等, 近年来研究较多的是全局连续的填充算法, 如连续 Fermat 螺线填充算法^[5]等. 但是, 针对边界特征比较复杂的模型, 目前的算法均存在打印效率低、打印耗材会有不同程度的浪费, 并且会产生过填充和欠填充等问题, 进而影响打印效果. 如何在保证边界细节特征的前提下, 最大限度地提升打印效率有着重要的研究意义.

本文研究分形这类边界细节特征比较复杂的模型的 3D 打印路径规划问题. 近年来, 分形在许多工程领域得到越来越多的应用, 分形艺术中优美丰富的图形可以应用到各种布局中, 如舞台设计、园林设计、建筑设计等. 利用分形很好的装饰性, 工程师设计比较有个性且优美的分形装饰品, 如分形画、分形灯饰等. 分形不仅展示了数学之美, 也揭示了世界的本质. 分形几何是真正描述大自然的几何学, 如海岸线、闪电和云朵等都是分形的杰作. 分形是科学与艺术的融合, 数学与艺术审美的统一. 因此, 如何将 3D 打印与分形造型相结合具有较重要的研究意义.

1 背景及相关工作

1.1 分形

分形是指具有自相似性的几何图形, 它们在工业设计领域和自然界中是普遍存在的, 如图 1 所示的蕨类植物的树叶、雪花和花朵都有分形元素.

数学分形产生的经典方法包括 L 系统、IFS 迭代函数系统和复变函数迭代系统等^[6-7].

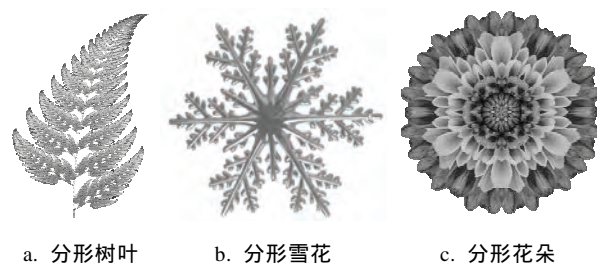


图 1 分形与自然

L 系统是分形生成的重要方法之一. L 系统实际上是字符串重写系统, 它把字符串解释成图形, 从一个生成元出发, 并按一定的规则, 通过对“F”, “+”, “-”等字符组成的字符串进行迭代替换生成分形. 图 2a 给出由经典的 L 系统生成的 Koch 雪花曲线.

Mandelbrot 集是由复平面上的二次多项式函数 $f(z) = z^2 + c$ 迭代生成, 其中, $c = a + bi$ 为复参数, a 和 b 分别为 c 的实部与虚部. 对于每一个参数 c , 从 $z_0 = 0$ 开始, 利用 $f(z)$ 进行迭代产生复数序列 $z_{n+1} = f(z_n)$, $n = 0, 1, \dots$; 则使得复数序列 $\{z_n\}$ 不发散至无穷的所有复数 c 所对应的复平面上的点集合组成 Mandelbrot 集.

固定参数 c , 以复数 $z = z_0$ 为初值, 利用 $f(z)$ 进行迭代产生的复数序列 $\{z_n\}$ 不发散至无穷的复数 z_0 所构成的集合为 Julia 集, 其随 c 的变化而变化.

图 2b, 2c, 2d 所示为 Julia 集取不同的参数时迭代 10 次所产生的图形边界, 图 2e 为迭代 10 次的 Mandelbrot 集边界.

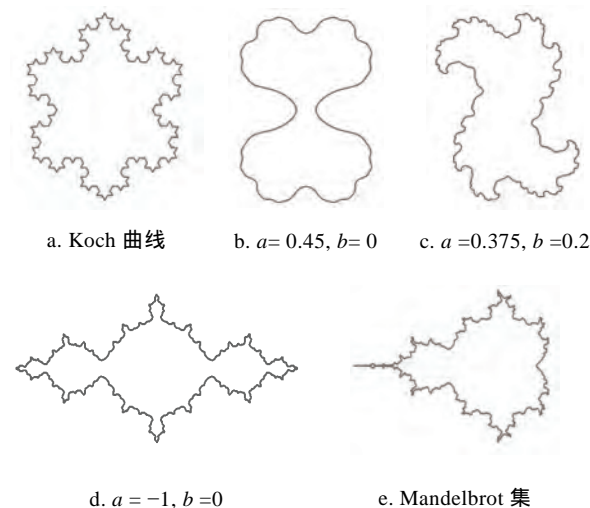


图 2 不同参数下的分形

1.2 3D 打印路径规划相关工作

目前针对 3D 打印路径规划的算法有很多, 经典

的路径规划算法有 zigzag 算法^[3,8-9], 轮廓平行扫描算法^[4,10-11]. 后来逐渐出现基于 Voronoi 图的扫描算法^[12]和星形发散扫描算法^[13]等. 近年来, 全局连续的填充路径成为 3D 打印的热门研究问题. 下面对上述算法进行简要介绍.

1.2.1 平行扫描

平行扫描又称为 zigzag 扫描, 每一段路径相互平行, 在边界线内往复扫描, 其以简洁易操作的优点应用最为广泛. Park 等^[8]利用加工区域的形状特征和路径区域选择最优打印倾角, 计算并存储分段路径最终连接平行路径, 从而得到全局 C^0 连续的路径. Rajan 等^[9]针对输入的模型建立以倾角为自变量, 平行路径的段数为目标的优化问题, 得出合理的扫描倾斜角; 虽然打印效率有所提升, 但要求打印模型的边界必须为直线或者圆弧.

在 3D 打印的实际过程中, 由于模型表面与打印方向有一定的角度且扫描线有一定的宽度, 使得物理层的精度受限, 实际成型轮廓与曲线轮廓之间会出现如图 3 所示误差, 称为台阶效应. 根据不同的精度, 会产生欠填充和过填充 2 种误差.

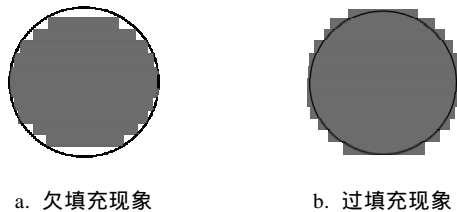


图 3 台阶效应

利用 zigzag 扫描算法在扫描过程中会产生较多的急转弯点和分割线, 从而影响打印效率. 在分割间断处打印机的喷嘴会产生抬起、快速定位和落下的过程, 这会增加打印时间. 打印过程中产生的拉丝现象, 会影响整体的打印效果.

1.2.2 轮廓平行扫描

轮廓平行扫描算法在现有的扫描算法中几何精度最高, 同时避免了填充截面时产生的台阶效应. Yang 等^[4]提出了一种高效计算等距线的算法, 其针对产生等距线的特殊情况(如自交)给出了相应的解决办法, 并通过与传统的等距线产生算法进行比较, 展示出算法的优越性. Farouki 等^[10]针对任何边界的轮廓平行等距线的生成算法, 通过实验表明其具有很好的鲁棒性. Jin 等^[11]提出一种自适应的扫描填充路径算法, 为保证原始模型的几何精度, 利用非均匀有理 B 样条表示切片层的

边界, 并产生向形心收缩的等距线. Tarabani^[14]在利用熔融沉积(fused deposition modeling, FDM)形式打印 3D 物体时, 提出基于最小支撑的轮廓平行填充路径优化算法, 其对实际打印时出现的多种情况都可保证打印的高效性.

由于是针对边界的偏置操作, 每条相邻的等距线并不相连. 目前越来越多的学者采用基于轮廓平行扫描算法构建全局连续算法, 大大降低了打印机喷嘴的升降次数, 在节省打印时间的同时, 提高了打印效率. 由于轮廓平行扫描与边界的特征紧密相连, 当边界较为复杂时, 内部填充曲线出现许多拐点并会产生如图 4 所示细小的打印碎片, 这不仅会浪费大量的打印时间, 也会造成打印材料的浪费.

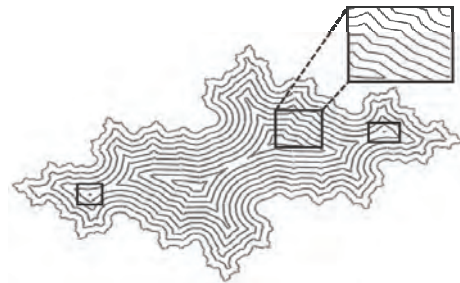


图 4 轮廓平行算法

1.2.3 空间填充曲线扫描

空间填充曲线有很多种, 如 Piano 曲线、Hilbert 曲线和螺旋线等. 空间填充曲线在打印区域内都是全局连续的, 但是其对于边界比较复杂的模型填充效果普遍并不理想. Revital 等^[14]基于文本的图像扫描技术提出一种空间填充曲线, 结果显示, 针对单连通区域, 相比 Piano-Hilbert 曲线, 其提升了图像的自相关性, 同时有利于图像与视频的压缩. Pedersen 等^[15]针对迷宫问题进行探讨, 提出基于迷宫结构的空间填充曲线, 并在许多方面有了实际的应用, 但是其在 3D 打印中会产生不必要的拐角, 进而降低了打印效率. Held 等^[16]提出利用一条 Fermat 螺线进行区域填充, 但此时螺线变形严重不利于实际打印. 之后又提出针对任意单连通的 2D 边界产生螺旋路径的算法^[17], 通过向区域内部中轴线上放置圆盘, 不断增大圆的半径, 并在边界处进行变形, 以达到区域填充的效果, 它可在产生螺旋线的过程中有效地避免自交, 并使得整体路径达到 C^2 连续. 但是, 该算法产生的螺旋线会出现过多的欠填充现象. 赵海森等^[5]提出空间填充曲线即连续 Fermat 螺线(connected Fermat

spirals, CFS), 其特点是连续且低曲率. 它将单连通区域分成若干子区域, 并在每个子区域中进行 Fermat 螺线连接, 最终将每个子区域连接. 但是, Fermat 螺线的保持特征性质较弱, 无法保证复杂边界的打印精度. 图 5 所示为 3 种不同空间填充曲线.

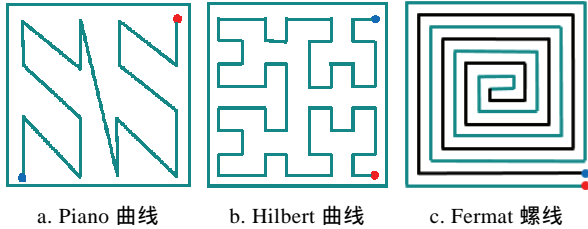


图 5 不同空间填充曲线

目前对分形模型的打印路径规划少有研究. Chiu 等^[18]针对每一层分形模型的切片建立网格, 利用若干像素点模拟分形模型的边界, 避免了分

形的欧几里得几何表示. 但是, 该算法侧重的是对分形边界的优化设计, 对内部填充曲线的路径规划算法并没有具体设计. 本文以分形模型的高效打印为出发点, 重点说明在 3D 打印过程中分形模型内部填充曲线全局连续的构造和优化算法.

2 本文算法

本文针对分形模型进行打印, 并提出一种新的 3D 打印路径规划算法. 为了保证边界精度和打印效率, 本文算法将区域分为轮廓层、过渡层及内层. 综合采用轮廓平行算法和 CFS 填充算法, 针对不同区域的特点采取不同的填充方式, 并对全局填充曲线进行了空间位置与角度的优化. 图 6 以 Mandelbrot 集为例给出本文整体算法流程.

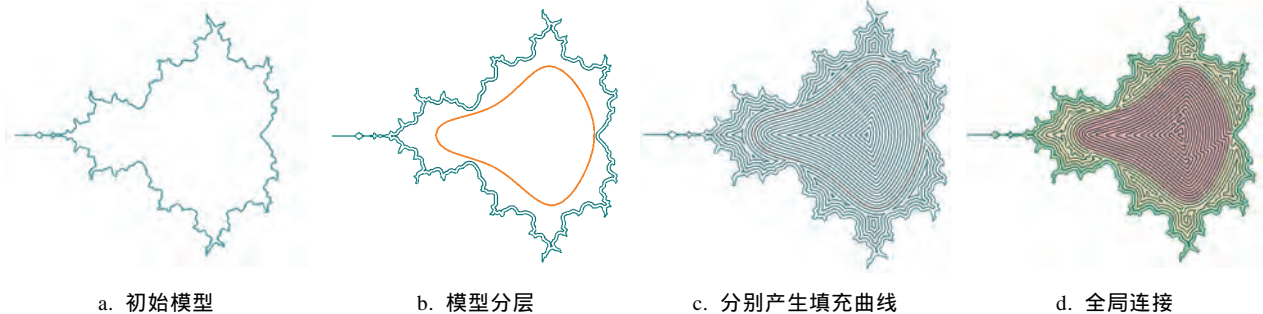


图 6 本文算法流程

2.1 轮廓层路径填充

本文利用轮廓平行算法填充轮廓层, 以达到保持边界特征的目的. 为确定轮廓层的厚度, 从模型的边界开始, 通过偏置操作产生一系列等距线. 对于等距线上的每一点检查其曲率信息, 若曲率大于某一阈值则视其为特征点. 特征点的分布能够体现整条等距线的主要特征和走向. 当某条等距线上的特征点数量降低至原始特征点数量的某个比例(本文设定为 50%)时, 停止等距线的生成, 即确定了轮廓层的厚度.

然而, 针对不同的边界也会出现特征点数量不变的情况(如 Koch 曲线). 如果等距线上的特征点数量不减少, 则利用面积进行优化; 本文设置轮廓层的面积小于总面积的 10%. 通过上述 2 种算法可得到轮廓层的最优层厚. 图 7 所示为轮廓层生成示意图.

轮廓层内各条等距线是互相分离的, 在实际打印过程中打印机的喷头存在上升与下降过程,

会造成时间浪费. 因此, 构造连续的轮廓层打印路径是有必要的. 本文采用最近邻扫描的算法, 扫描完成每条等距线后, 搜索其余未扫描等距线的最近点进行扫描, 最终得到连续的轮廓层填充曲线.

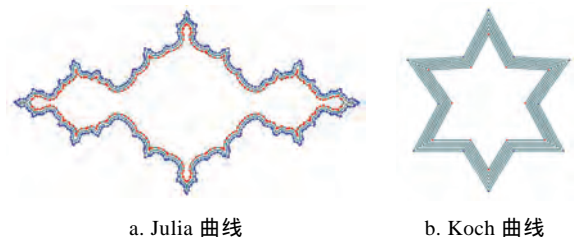


图 7 不同曲线轮廓层生成

2.2 过渡层优化问题

2.2.1 优化过渡层层厚

本文的目的是在保证一定精度的前提下最大限度地提升打印效率, 若过渡层太厚, 会影响整体打印效率; 否则, 在利用三次 B 样条拟合过渡层的内边界时会因为过渡层的内外边界出现相交情况,

进而影响打印效果. 为此, 利用过渡层的面积为初步目标, 设置过渡层的面积的下限 S_{\min} (本文设为原模型总面积的 30%), 从轮廓层内边界开始, 不断进行等距偏置操作; 当过渡层的面积大于 S_{\min} 时, 则停止偏置操作, 取最后一条等距线为过渡层内边界的初始边界线. 该曲线也是内层的初始外边界曲线.

2.2.2 初始边界线预处理

由于等距线在进行偏置的过程中仍然具有较多的细节, 因此需要将偏置后得到的过渡层内边界的初始边界曲线的细节进行简化, 以便进行后续的三次 B 样条拟合计算. 首先将初始边界曲线像素化, 确定覆盖过渡层初始内边界的像素多边形, 提取每个像素的中心点如图 8a 所示. 其次剔除不必要的像素点, 使得所有的像素中心点连接成一条能够遍历所有像素的最大图 $G=(V, E)$; 其中 V 表示像素中心点, E 表示边(在此规定边是水平和垂直方向). 剔除策略如下:

Step1. 利用 $\deg(V)$ 表示像素中心点 V 的度, 删除 $\deg(V)=1$ 的像素(如图 8b 中情况 i 所示).

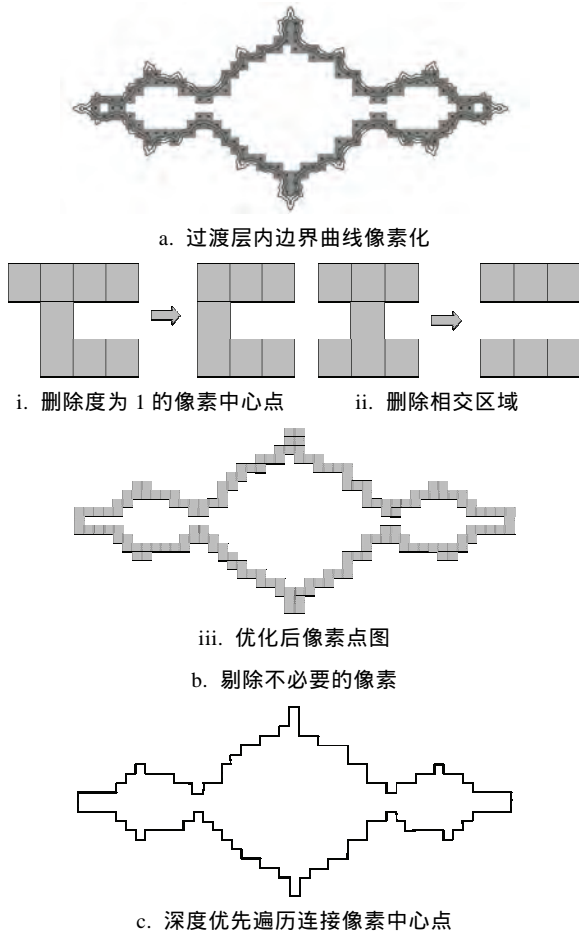


图 8 过渡层内边界初始曲线的预处理过程

Step2. 判断相交区域, 删除‘H’或‘工’字的中间相交部分(如图 8b 中情况 ii 所示), 优化后的过渡层内边界的像素点见图 8b 中 iii 所示.

Step3. 将像素中心点按照深度优先遍历原则进行连接, 结果如图 8c 所示, 大大减少了数据存储空间.

2.2.3 三次 B 样条拟合过渡层内边界

预处理后的过渡层内边界并不光滑, 直接进行偏置操作依旧会产生过多的拐点, 进而影响打印效率. 为了确定过渡层内边界的光顺曲线, 利用三次 B 样条进行拟合^[19-20]. 采用 B 样条曲线拟合的好处有: 利用 B 样条的局部影响性质, 抑制噪声点对整条边界线的影响, 使曲线趋于光滑; 进一步减少数据量, 使边界线的表示更加紧凑, 有助于内部填充曲线的计算.

(1) 优化目标函数

本文利用三次 B 样条对过渡层初始内边界进行拟合优化, 设最终拟合曲线的表示为

$p(t) = \sum_{i=1}^n P_i \cdot B_i^3(t)$. 其中, $p(t)$ 为三次 B 样条, $P_i (i=1, 2, \dots, n)$ 是控制多边形的顶点; $B_i^k(t)$ 是 k 次 B 样条基函数, 且以 $U = \{u_1 \leq \dots \leq u_{k+1} \leq \dots \leq u_{n-k} \leq \dots \leq u_{n+k+1}\}$ 为节点向量, $n \geq 2k+1$. $B_i^k(t)$ 递归定义为(k 为 B 样条的次数)

$$B_i^0(t) = \begin{cases} 1 & u_i \leq t \leq u_{i+1} \\ 0 & \text{其他} \end{cases}$$

$$B_i^{k+1}(t) = \frac{t-u_i}{u_{i+k}-u_i} B_i^k(t) + \frac{u_{i+k+1}-t}{u_{i+k+1}-u_{i+1}} B_{i+1}^k(t)$$

为得到光滑的过渡层内边界曲线, 建立优化

问题 $\min \sum_{i=0}^m (p(t_i) - q_i)^2 + \lambda \int_0^1 |p''(t)|^2 dt$. 其中, $\{q_i\}$

($i=0, 1, \dots, m$) 为过渡层初始内边界上的采样点; $\{t_i\}$ ($i=0, 1, \dots, m$) 为对应的参数, 可利用累积弦长参数化算法计算获得. 目标函数的第 1 项为拟合项, 即初始边界线上的拟合点与光滑曲线上对应点的最小二乘距离; 第 2 项为光滑项, 取为光滑曲线的曲率的平方积分; λ 为平衡参数, 本文取 $\lambda = 0.03$. 通过该优化问题求解最佳拟合三次 B 样条的控制点坐标 P_i , 即得最佳拟合曲线.

(2) 节点向量的确定

本文利用累积弦长参数化的算法对数据进行参数化, 具体的参数值确定公式为

$$L = \sum_{i=1}^m |q_i - q_{i-1}|$$

$$t_0 = 0, t_i = t_{i-1} + \frac{|q_i - q_{i-1}|}{L}, i = 1, 2, \dots, m$$

将 $m+1$ 个参数值合理分配到定义域的 $n-k+1$ 个内节点区间内, 采用 $d = \frac{m+1}{n-k+1}$, $i = \text{int}(jd)$, $\alpha = jd - i$, $u_{k+j} = (1-\alpha)t_{i-1} + \alpha t_i$, $j=1, 2, \dots, n-k$ 确定节点向量.

按照上述算法构成的节点向量可以保证每个内节点区间至少包含一个参数值 t_i . 由于拟合曲线为闭曲线其节点向量和控制顶点分别满足周期性条件

$$\begin{cases} u_{i+n-k} = u_i + 1, (i=1, \dots, k) \\ u_{i+n-k+1} = u_{i+1} + 1, (i=k+1, \dots, 2k) \\ u_{k+1} + 1 = u_{n+1} \\ P_{l+j} = P_j (j=1, \dots, k; l=n-k) \end{cases}$$

(3) 光顺项处理

针对三次 B 样条求解 2 次导数得到分片一次多项式 $p''(t)$, 计算 $p''(t)$ 在 $[0,1]$ 的积分, 即

$$\int_0^1 p''(t) dt = \int_{u_4}^{u_5} p''(t) dt + \int_{u_5}^{u_6} p''(t) dt + \dots + \int_{u_{n-1}}^{u_n} p''(t) dt.$$

利用 2 点 Gauss 积分

$$\int_a^b p''(t) dt = \frac{b-a}{2} \cdot \left(a_1 \cdot p''\left(\frac{a+b}{2} + \frac{b-a}{2} \cdot t_1\right) + a_2 \cdot p''\left(\frac{a+b}{2} + \frac{b-a}{2} \cdot t_2\right) \right)$$

进行简化计算. 其中, $a_1 = a_2 = 1$, $t_1 = 0.57735$, $t_2 = -0.57735$. a 和 b 表示参数 t 所在区间的上下限.

(4) 构造方程组求解

最终的优化问题归结为求解以控制点为未知数的线性方程组问题. 由于 B 样条的局部性, 系数矩阵是稀疏的, 因此可用 Eigen 库中求解稀疏矩阵的算法进行快速计算. 对拟合后的曲线与过渡层外边界进行相交检测, 如果相交, 则向中心偏置, 直至无交点. 由此进一步确定了最优的过渡层厚度. 图 9 所示内部闭合光顺曲线为求得的过渡层内边界(即内层边界)的拟合曲线.

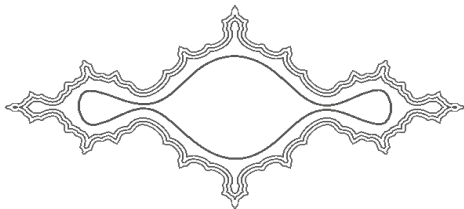


图 9 过渡层内边界拟合曲线

2.3 CFS 填充

2.3.1 全局连接算法

本文在赵海森等^[5]工作的基础上将过渡层和

内层的填充曲线进行连接. 首先对过渡层与内层进行一系列的等距线偏置操作, 初步得到过渡层与内层的填充路径. 以过渡层与内层等距线为节点, 建立初始图 G_{initial} . 首先针对等距线进行编号 $l_{i,j}$, 其中, i 表示当前等距线到边界 ∂R 的距离 $d(\partial R, l_{i,j}) = i \cdot d$, 且 d 为偏置距离, j 表示到边界相同距离的等距线的序号, $l_{0,1}$ 总为最外层边界.

连接相邻等距线 $l_{i,j}$ 和 $l_{i+1,j'}$, 确定待连接区域 $O_{i,j,j'}$, 其定义为

$$O_{i,j,j'} = \{p \in l_{i,j} \mid d(p, l_{i+1,j'}) < d(p, l_{i+1,k}), k \neq j'\}.$$

其中, $d(p, l)$ 为点 p 到等距线 l 的距离. 如果 $O_{i,j,j'}$ 不为空集, 则在 $l_{i,j}$ 和 $l_{i+1,j'}$ 之间加入一条边, 权重设置为 $O_{i,j,j'}$ 中点的距离.

将初始图 G_{initial} 转化为最小生成树, 以 $l_{0,1}$ 为根, 树的节点分为 2 种类型, 即节点的度小于等于 2 与节点的度大于 2. 从叶子节点遍历至根节点. 重新进行路径规划, 得到图 10 所示全局连接曲线.



图 10 初步全局 CFS

2.3.2 局部曲线优化算法

目前利用 Fermat 螺线已经将过渡层和内层连成一条线, 但是仍存在大量拐点和相邻等距线过近的情况. 本文提出一种局部曲线优化的算法, 针对曲率较大的拐点和等距线距离过近(远)的情况进行局部优化.

首先进行局部曲率优化. 检测 CFS, 当前点 Q_i 处的角度在 $(30^\circ, 120^\circ)$ 区间内时, 提取样本点 (一般取为当前点的前后 2 个点 Q_{i-2} 和 Q_{i-1} , Q_{i+1} 和 Q_{i+2}) 利用二次样条曲线进行曲率优化. 当角度小于 30° 时, 建立角度与最终映射点的关系

$$Q_i = ((Q_{i+1} + Q_{i-1}) / 2 + Q_i) / 2.$$

将原始位置映射到上述位置, 尽可能地使整体曲率降低. 由于内层边界利用光顺三次 B 样条逼近, 内层产生的等距线较光滑, 但是在部分区域仍然存在急转弯点. 利用本文算法可大大降低

急转弯比例, 图 11 所示 a_1 和 a_2 为局部曲率优化效果对比.

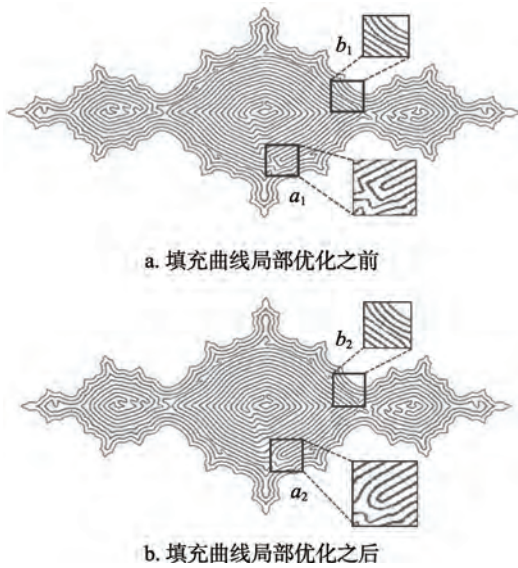


图 11 填充曲线局部优化前后对比图

为了达到良好的视觉效果, 避免实际打印时过填充与欠填充问题, 保证填充线均匀分布是关键. 经过分析, 等距线过近的情况主要存在于最后一条等距线以及不同区域相邻的等距线.

首先检测等距线中的相邻区域, 图 12 所示为局部位置优化的步骤. l_1, l_2 表示不同区域的边界, 当 2 条边界的距离小于 d (偏置距离) 时, 确定小于 d 的 2 个部分, 并分别针对这 2 部分等距离选取样本点. p_0 和 p_{m+1} 表示 l_1 上的临界的原始点, 在等距线 l_1 上的样本点为 p_1, \dots, p_m ; q_0 和 q_{m+1} 表示 l_2 上的临界的原始点, 在等距线 l_2 上的样本点为 q_1, \dots, q_m . 确定每个样本点的法向方向 n , 向外法向方向偏置一定的距离 Δd , 分别得到 p'_1, \dots, p'_m 与 q'_1, \dots, q'_m 2 组样本点的平移点. 最后分别针对 $p_0, p'_1, \dots, p'_m, p_{m+1}$ 和 $q_0, q'_1, \dots, q'_m, q_{m+1}$ 建立三次样条拟合开曲线, 从而避免了因等距线上原始点的平移而导致的平滑问题, 图 11 中 b_1 和 b_2 显示了局部位置优化效果对比. 针对一条等距线上距离较近的 2 部分优化同理.

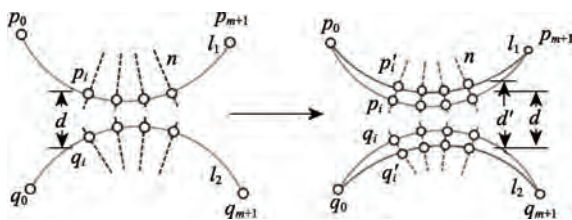


图 12 局部位置优化示意图

确定外法向偏置距离 Δd : 将 l_1, l_2 中距离较近的 2 部分上的样本点配对, p_i 与 q_i 对应; 然后利用 $\Delta d = d' - d(p_i - q_i)$ 得到每对点应该偏置的距离. 通过上述算法得到图 11 改善效果图, 其相比于优化之前减少了过填充, 其中红色方框部分为相比之前更优位置结果.

2.4 多连通区域

在工程应用中, 3D 打印切片多为多连通区域. 本文方法经过适当修改也可以打印多连通区域. 采取在分形内部嵌套分形的形式构造多连通区域, 按照第 2.1~2.3 节算法, 依然将打印区域分为轮廓层、过渡层和内层. 在建立过渡层时需要产生 2 条或多条(根据模型的亏格数而定)光滑三次 B 样条曲线. 然后利用 Fermat 螺线填充算法进行连接, 并进行局部曲线的位置和曲率优化. 图 13 所示为分形内部挖去若干分形所构成区域的 3D 打印路径.

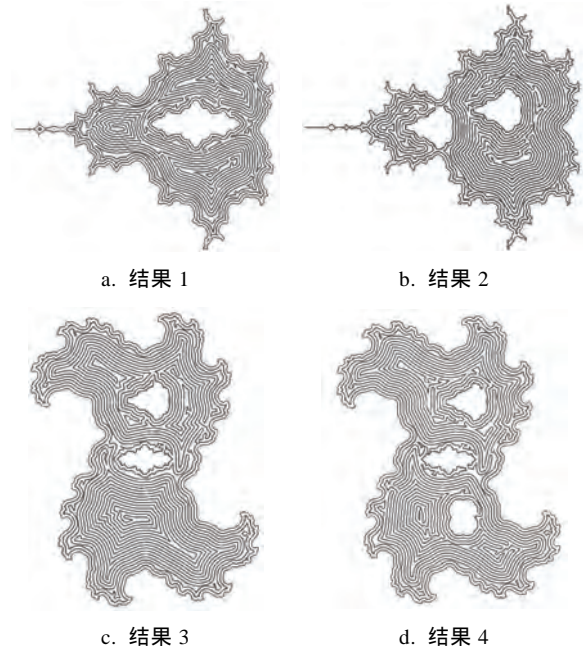


图 13 多连通区域的填充曲线

从实验结果看出, 本文算法对多连通区域仍然适用; 不过由于区域变得复杂, 模型打印效率随之降低, 且欠填充的比例和急转弯比例有所增加.

3 实验结果

本文从 5 个影响因素分析算法的优劣, 即填充分割段数(number of segments, SEG), 路径角度(急转弯的数量(sharp turn points, ST)和最小角度

(minimum angle, MA), 欠填充(under-fill, UF)与过填充(over-fill, OF)比例, 打印成本(时间成本(printing time, T)和材料成本(printing material, M)和视觉效果. 利用传统的 zigzag 算法(zigzag, Z)^[3], 轮廓平行算法(contour parallel, C)^[4], Fermat 螺线填充算法(connected Fermat spirals, CFS)^[5]与本文算法(our path, O)进行对比, 并利用 8 组分形模型进行测试.

3.1 3D 打印机运行环境与基本设置

本实验的打印机型号是 Ultimate 2+, 本文的自适应边界算法利用 C++ 进行实现, 运行环境为 64 位操作系统, Inter(R) Core(TM) i7-4790 CPU @ 3.60GHz 处理器, 8GB 内存.

基本设置为喷嘴直径为 0.4mm, 每层高度 0.2mm, 最大的扫描速度是 70mm/s.

本文选取 8 组数据, 用序号 1~8 表示测试图形. 1~5 分别表示 Julia 集分别取不同参数时的图形; 1. $a = 0.2, b = 0.5$; 2. $a = 0.3, b = 0$; 3. $a = 0.68, b = 0.305$; 4. $a = 0.375, b = 0.2$; 5. $a = -1, b = 0$; 6 表示 L 系统分形; 7 表示 Koch 曲线; 8 表示 Mandelbrot 集迭代 10 次所产生的曲线.

其中符号#表示数目, %表示比例, &表示($^{\circ}$). 下文中, Z, C, CFS 和 O 分别表示 zigzag 算法、轮廓平行扫描算法、Fermat 螺线算法和本文算法.

3.2 填充分割数

经过测试, 统计 4 种算法所产生填充曲线的分割数如表 1 所示. 可以看出, zigzag 路径的分割数最多, 其次是轮廓平行算法; CFS 算法和本文算法达到全局连续, 故分割数为 1.

表 1 算法分割路径数

分形	#Z _{SEG}	#C _{SEG}	分形	#Z _{SEG}	#C _{SEG}
1	71	12	5	80	20
2	70	15	6	92	15
3	85	13	7	97	17
4	87	14	8	76	15

3.3 急转弯数量

急转弯的比例直接影响打印效率, 为计算急转弯比例, 本文针对填充曲线均匀选取样本点, 分别计算样本点处的角度. 本文选取 50 000 个样本点, 当样本点处的角度小于 100° 时判定为急转弯; 之后统计样本点的角度, 并计算急转弯的比例.

针对 8 组不同的分形图案测试 4 种算法的急转弯比例和最小角度, 结果如表 2~3 所示. Fermat 螺线填充是在轮廓平行的基础上进一步优化, 急转

弯比例相对减小; 本文利用三次 B 样条光顺产生螺线可以有效地减少急转弯比例, 且在 4 种算法中急转弯比例最低.

表 2 算法急转弯比例比较

分形	%Z _{ST}	%C _{ST}	%CFS _{ST}	%O _{ST}
1	2.24	2.08	1.81	1.62
2	1.60	3.02	1.34	1.16
3	2.92	1.90	1.86	1.58
4	1.68	2.04	1.74	1.45
5	3.20	1.44	1.39	0.92
6	0.28	0.49	0.41	0.32
7	2.12	2.36	2.09	1.53
8	2.16	2.00	1.45	1.43

表 3 给出最小角度的比较结果表明, 本文算法可以有效地提升最小角度, 并将最小角度平均提升 18.19° , 避免了角度过小影响打印效率问题. 说明: zigzag 算法相邻填充线平行夹角为 0° .

表 3 算法最小角度比较

分形	&Z _{MA}	&C _{MA}	&CFS _{MA}	&O _{MA}
1	0	3.31256	25.3684	33.5821
2	0	8.56324	15.2861	21.1282
3	0	0.16930	15.3567	18.5269
4	0	4.86297	35.4681	25.3647
5	0	7.58557	26.3519	28.1274
6	0	6.23350	23.5827	25.1286
7	0	2.37124	10.5892	13.5675
8	0	3.22980	33.2571	36.4297

3.4 填充(欠填充与过填充)比例

填充比例不仅是影响打印效率的因素, 而且直接影响了打印效果. 当欠填充比例较高时, 模型内部产生空隙; 当过填充比例较高时, 模型内部填充材料堆积会造成填充不均, 影响打印的美观性.

为了定量计算欠填充与过填充的比例, 本文采取将线材直径加大的算法, 计算过填充区域与欠填充区域占有所有填充区域的面积. 图 14 所示龙曲线按照 4 种算法所得欠填充与过填充的效果图.

计算 4 种算法在 8 个例子中的欠填充与过填充比例, 结果如表 4 和表 5 所示; 将其中数据转化为直方图分别为图 15 和图 16 所示.

从图 15 可看出, zigzag 算法欠填充比例最低, 而 Fermat 螺线填充算法欠填充比例最高; 本文算法的欠填充比例比轮廓平行算法和 Fermat 螺线填充算法低, 比 zigzag 算法高 0.08%.

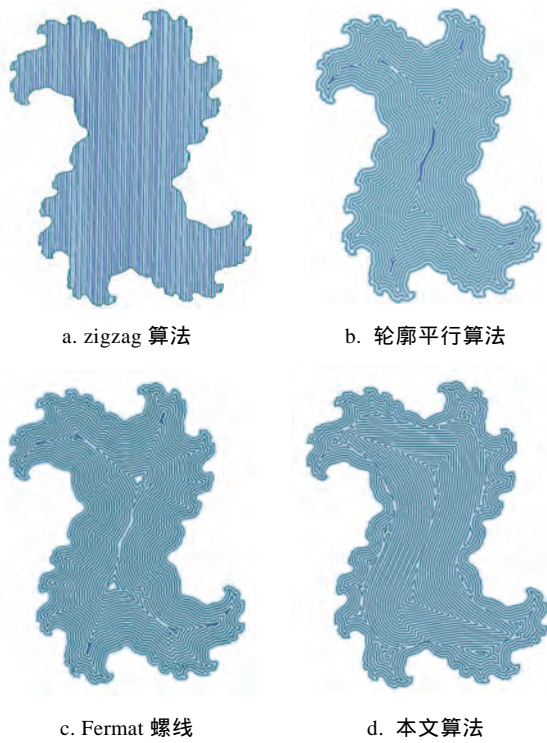


图 14 龙曲线欠填充与过填充

表 4 欠填充比例 %

分形	%Z _{UF}	%C _{UF}	%CFS _{UF}	%O _{UF}
1	0.256897	0.478253	0.647826	0.385462
2	0.145689	0.172237	0.285431	0.278613
3	0.295436	0.364830	0.354169	0.219841
4	0.357825	0.529137	0.682174	0.369824
5	0.145236	0.164787	0.468423	0.214863
6	0.216834	0.392861	0.598214	0.452987
7	0.214567	0.413658	0.519672	0.349657
8	0.298562	0.312563	0.349821	0.295413

表 5 过填充比例 %

分形	%Z _{OF}	%C _{OF}	%CFS _{OF}	%O _{OF}
1	0.835612	0.298827	0.145268	0.219852
2	0.794655	0.284701	0.231498	0.272586
3	0.793521	0.259527	0.235741	0.249162
4	1.009132	0.378642	0.398526	0.413985
5	0.865127	0.433155	0.356985	0.386954
6	0.731864	0.354821	0.259874	0.315934
7	0.793541	0.265213	0.217543	0.268517
8	0.893619	0.394281	0.378541	0.415975

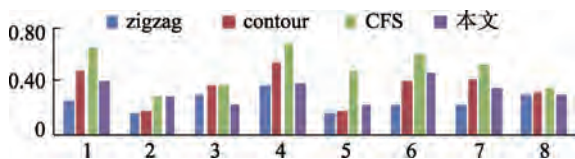


图 15 欠填充比例直方图

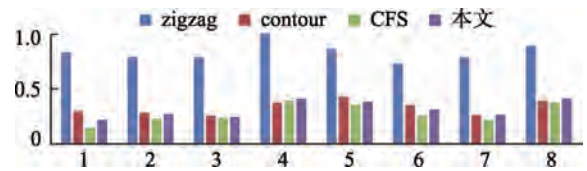


图 16 过填充比例直方图

图 16 表明, 本文算法过填充的比例有小幅度增加, 针对 8 组数据, 本文算法的过填充比例平均值比轮廓平行算法和 zigzag 平行算法低, 比 Fermat 螺线填充算法高 0.04%。

3.5 打印成本

减少打印成本、提升打印效率是 3D 打印的重要目标. 针对相同填充率, 利用 4 种算法打印, 并统计 8 种模型用不同的打印算法的时间成本与材料消耗, 得到表 6 和表 7 所示的结果。

表 6 打印时间比较 s

分形	#Z _T	#C _T	#CFS _T	#O _T
1	472	512	452	414
2	516	486	428	421
3	534	563	503	473
4	687	663	624	616
5	470	509	415	408
6	501	469	421	417
7	517	476	403	391
8	566	545	519	428

表 7 打印耗材比较 mm

分形	#Z _M	#C _M	#CFS _M	#O _M
1	2336.83	2255.85	2058.16	1989.26
2	2913.17	2839.45	2651.32	2593.89
3	3454.59	3366.06	3251.64	3053.46
4	3714.83	3624.55	3369.51	3321.59
5	2868.61	2788.08	2589.63	2510.65
6	2095.63	2062.98	1869.43	1872.52
7	3079.60	2971.72	2898.21	2569.39
8	3313.40	3225.17	2832.16	2945.35

本文算法在时间和材料成本都是最低的. 经过进一步的统计分析, 在打印时间方面, 本文算法较 zigzag 算法平均节省了 16.41%, 较轮廓平行算法平均节省了 15.73%, 较 Fermat 螺线填充算法平均节省了 5.1%. 由于急转弯的比例低, 因此降低了打印时间; 材料成本相对于 zigzag 算法平均降低了 13.06%, 较轮廓平行算法平均降低了 7.1%; 较 Fermat 螺线填充算法平均节省了 3.7%. 本文算法打印的整体工作效率从时间和耗材 2 个方面

都得到了明显的提升，并且针对复杂的轮廓模型具有更大的优势。

3.6 视觉效果

通过观察实际打印结果对比可以看出，使用 zigzag 路径整体打印结果平整，出现部分拉丝现象；在平行线与边界相交处会产生过填充问题，对于细节区域往往不能保证边界的精度；对于轮廓平行算法，在相邻的等距线之间打印喷头经历上升与下降的过程会产生过填充现象；Fermat 螺线填充算法在急转弯处出现欠填充现象。

图 17 所示为利用 Julia 集进行测试得到 zigzag 打印结果，红色方框 1 标出 zigzag 打印路径和模型外轮廓在快速打印过程中产生的欠填充问题，红色方框 2 表示 zigzag 路径在轮廓处产生的过填充问题。

对比图 18 所示 2 种算法产生的边界和内部填充区域，Fermat 螺线填充算法在打印时依旧保持边界的转角特征使得方框 1 中填充曲线拐角明显，而本文算法避免了这种情况的发生。这是因为无论轮廓多么复杂，总可以确定一个光滑的内层轮廓使得填充曲线在内部平滑，从而节省了打印时间和打印材料成本。

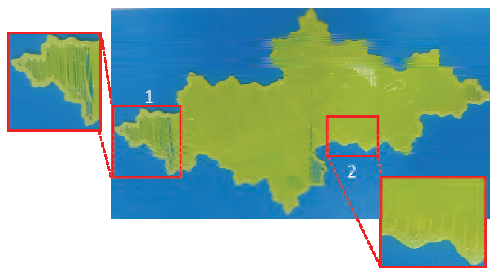


图 17 zigzag 路径打印问题



a. Fermat 螺线填充 b. 本文算法

图 18 Fermat 螺线填充与本文算法细节对比

本文算法产生的模型内部填充更加光滑、视觉效果更好。针对不同参数下的分形模型，更多打印结果如图 19~图 21 所示，图中红色方框为 Fermat 螺线填充与本文算法细节对比。

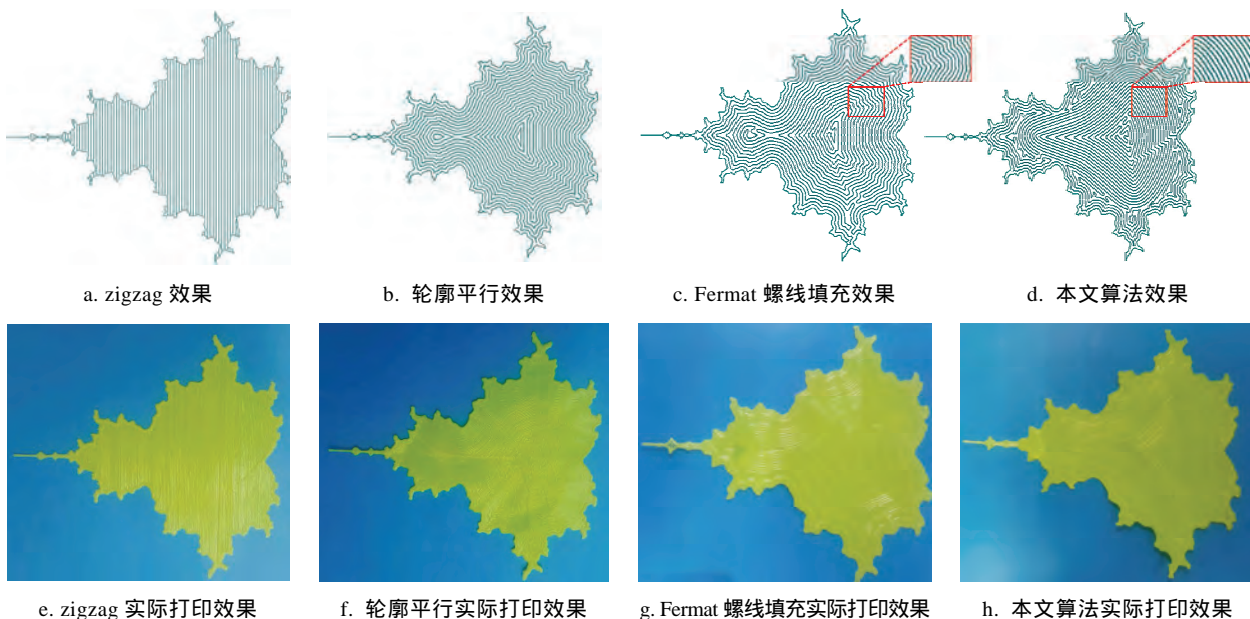
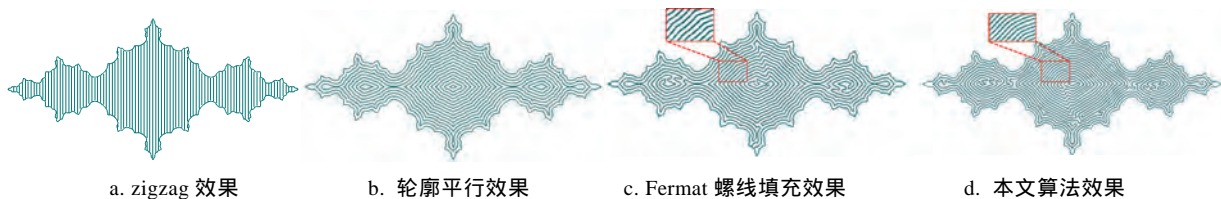
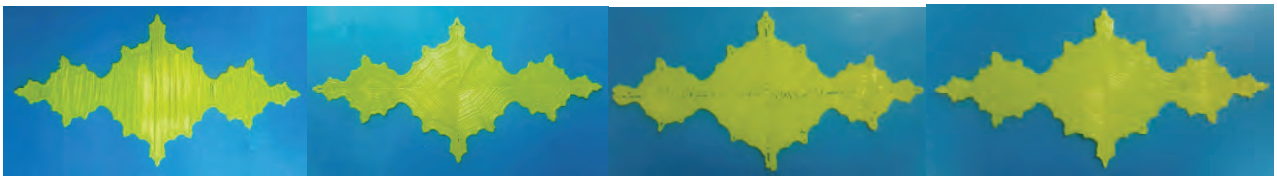


图 19 不同算法 Mandelbrot 集打印结果

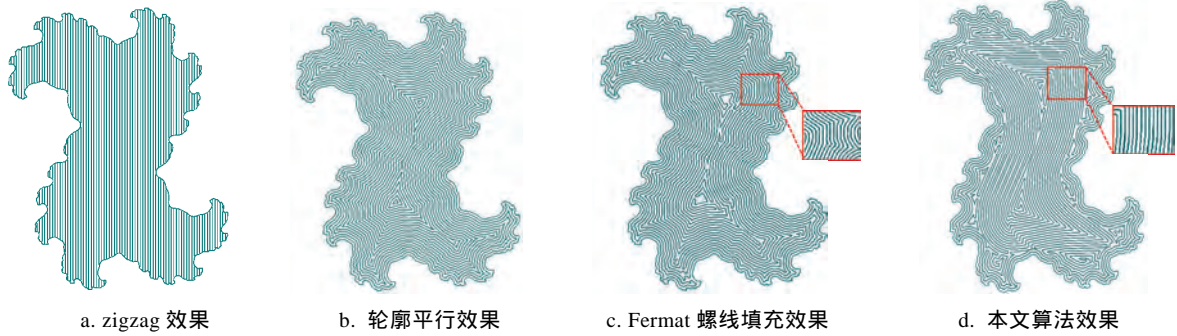


a. zigzag 效果 b. 轮廓平行效果 c. Fermat 螺线填充效果 d. 本文算法效果

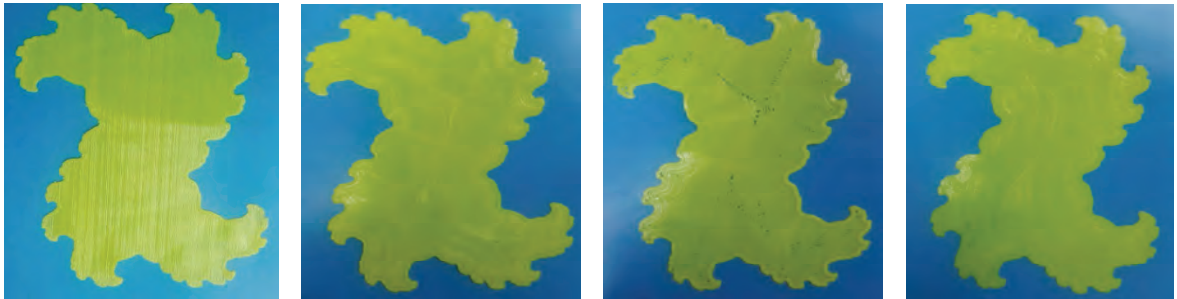


e. zigzag 实际打印效果 f. 轮廓平行实际打印效果 g. Fermat 螺线填充实际打印效果 h. 本文算法实际打印效果

图 20 不同算法 Julia 集($a=-1, b=0$)打印结果



a. zigzag 效果 b. 轮廓平行效果 c. Fermat 螺线填充效果 d. 本文算法效果



e. zigzag 实际打印效果 f. 轮廓平行实际打印效果 g. Fermat 螺线填充实际打印效果 h. 本文算法实际打印效果

图 21 不同算法龙曲线打印效果

4 分析与讨论

4.1 建立打印效率的数学模型

3D 打印路径规划的重点是如何节省 3D 打印的时间成本和材料成本. 本节对该问题的一般数学模型进行一些讨论.

4.1.1 时间成本

在打印材料成本一定的前提下, 影响实际 3D 打印时间的因素主要有分割数与急转弯比例.

(1) 分割数对时间成本的影响

在实际的打印过程中, 3D 打印机遇到两段分离的路径时, 打印机的喷头会产生上升、快速定位和下降的过程, 这个过程消耗较多的打印时间.

(2) 急转弯比例对时间成本的影响

在打印过程中, 遇到急转弯 $\angle O$ 时, 为保证精度, 打印机会在 A 处减速到 O , 返回 B 的途中加速. 如果平滑急转弯点 O , 可有效地避免打印机的减速加速过程造成的时间浪费, 如图 22 所示.

因此, 3D 打印时间成本有近似关系

$$f_{\text{time}} = \alpha_1 \cdot f_{\text{SEG}} + \alpha_2 \cdot f_{\text{ST}} + C_1.$$

其中, f_{time} 为打印时间成本; f_{SEG} 为以分割数为自变量的函数; f_{ST} 为以急转弯比例为自变量的函数; α_1 和 α_2 为权函数; C_1 为打印时间基数.

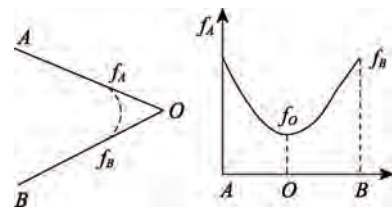


图 22 急转弯打印速度变化

4.1.2 材料成本

在切片层打印区域给定的前提下, 影响实际材料成本的因素为实际填充比例. 图 23 所示为打印路径走折线时过填充与欠填充区域的示意图. 其中, 虚线表示打印机扫描路径, 实线表示挤出材料边界, 内部填充区域为欠填充, 边缘填充区域为过填充.

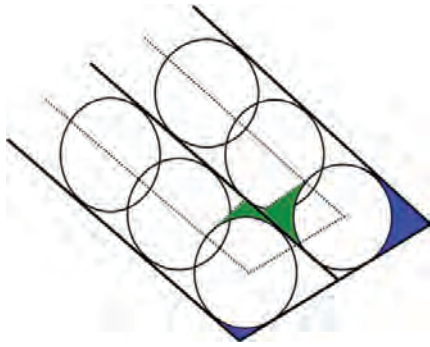


图 23 欠填充与过填充示意图

由于打印路径具有一定宽度, 在面对急转弯或者距离太近的路径打印时会产生过填充问题, 当设置的填充率过低时, 又会产生欠填充问题. 在实际打印过程中, 为保证打印效果, 通过增加整体填充率使得欠填充比例一般较低, 而过填充与欠填充的比例随着相邻路径角度减少而增加.

材料成本的近似关系可近似表示为

$$f_{\text{material}} = \beta \cdot f_{\text{over-fill}} + C_2.$$

其中, f_{material} 表示 3D 打印的材料成本; $f_{\text{over-fill}}$ 表示以过填充比例为自变量的函数; β 表示权函数; C_2 为所需材料基数.

因此, 总的打印成本可近似表示为

$$f_{\text{total}} = \omega_1 \cdot f_{\text{time}} + \omega_2 \cdot f_{\text{material}} = \omega_1 \cdot (\alpha_1 \cdot f_{\text{seg}} + \alpha_2 \cdot f_{\text{st}} + C_1) + \omega_2 \cdot (\beta \cdot f_{\text{over-fill}} + C_2).$$

其中, ω_1 和 ω_2 是 2 个常数.

在降低时间成本和材料成本的要求下, 如何进行合理的 3D 打印路径规划是值得探讨的课题.

4.2 本文算法的局限性

利用本文算法实际的打印结果显示, 所产生的填充曲线仍存在部分欠填充现象, 并且主要集中在过渡层. 一种解决的方法是增加打印密度, 但这又可能带来过填充的现象, 浪费材料.

另外, 本文主要讨论以分形面为主的单连通模型路径规划. 虽然本文算法对多连通模型仍然适用, 但是分形的边界细节特征较多, 内部嵌套分形产生多连通区域增加了模型的复杂度; 处理该类复杂的多连通区域算法较为耗时, 且可能增加欠填充的比例.

5 结 语

随着 3D 打印技术的快速发展, 人们对打印的模型要求也越来越高. 本文针对分形面这类具有复杂边界的模型, 旨在保证精度的前提下改善填

充路径, 以提升 3D 打印的效率.

本文首先介绍了对于分形模型产生自适应边界的填充路径的步骤. 然后分别从填充路径的分割数、路径角度(急转弯比例与最小角度)、填充的均匀程度(过填充与欠填充比例)、打印成本(时间和耗材)和打印效果 5 个方面对本文算法进行验证, 得到了较好的结果. 本文算法不仅针对分形, 而且对任意不同复杂程度的边界模型都可以在保证边界特征的同时生成一条全局连续的填充曲线.

今后的工作将会针对过渡层的欠填充问题设计优化策略, 进一步改进填充曲线. 另外探讨如何有效地在复杂边界的多连通区域和分形线内生成连续的填充曲线也是值得进一步探讨的研究课题.

参考文献(References):

- [1] Liu Ligang, Xu Wenpeng, Wang Weiming, *et al.* Survey on geometric computing in 3D printing[J]. Chinese Journal of Computers, 2015, 38(6): 1243-1267(in Chinese)
(刘利刚, 徐文鹏, 王伟明, 等. 3D 打印中的几何计算研究进展[J]. 计算机学报, 2015, 38(6): 1243-1267)
- [2] Hu Ruizhen, Huang Hui. Recent progress in 3D printing inspired fabrication optimization[J]. Journal of Computer-Aided Design & Computer Graphics, 2015, 27(6): 961-967 (in Chinese)
(胡瑞珍, 黄惠. 3D 打印启发下的模型实例化优化研究综述[J]. 计算机辅助设计与图形学学报, 2015, 27(6): 961-967)
- [3] Ding D H, Pan Z X, Cuiuri D, *et al.* A tool-path generation strategy for wire and arc additive manufacturing[J]. The International Journal of Advanced Manufacturing Technology, 2014, 73(1-4): 173-183
- [4] Yang Y, Loh H T, Fuh J Y H, *et al.* Equidistant path generation for improving scanning efficiency in layered manufacturing[J]. Rapid Prototyping Journal, 2002, 8(1): 30-37
- [5] Zhao H S, Gu F L, Huang Q X, *et al.* Connected Fermat spirals for layered fabrication[J]. ACM Transactions on Graphics, 2016, 35(4): Article No.100
- [6] Liu Huajie. Fractal art[M]. Changsha: Hunan Science and Technology Press in Hunan, 1998(in Chinese)
(刘华杰. 分形艺术[M]. 长沙: 湖南科学技术出版社, 1998)
- [7] Zhang Jizhong. Fractal[M]. 2nd ed. Beijing: Tsinghua University Press, 2011(in Chinese)
(张济忠. 分形 [M]. 2 版. 北京: 清华大学出版社, 2011)
- [8] Park S C, Choi B K. Tool-path planning for direction-parallel area milling[J]. Computer-Aided Design, 2000, 32(1): 17-25
- [9] Rajan V T, Srinivasan V, Tarabanis K A. The optimal zigzag direction for filling a two-dimensional region[J]. Rapid Prototyping Journal, 2001, 7(5): 231-240
- [10] Farouki R T, Koenig T, Tarabanis K A, *et al.* Path planning with offset curves for layered fabrication process[J]. Journal of Manufacturing Systems, 1995, 14(5): 355-368

- [11] Jin G Q, Li W D, Gao L. An adaptive process planning approach of rapid prototyping and manufacturing[J]. *Robotics and Computer-Integrated Manufacturing*, 2013, 29(1): 23-38
- [12] Kim D S. Polygon offsetting using a Voronoi diagram and two stacks[J]. *Computer-Aided Design*, 1998, 30(14): 1069-1076
- [13] Onuh S O, Hon K K B. Application of the Taguchi method and new hatch styles for quality improvement in stereo lithography[C] // *Proceedings of the Institution of Mechanical Engineers*. London: Sage Publications, 1998: 461-472
- [14] Dafner R, Cohen-Or D, Matias Y. Context-based space filling curves[J]. *Computer Graphics Forum*, 2000, 19 (3): 209-218
- [15] Pedersen H, Singh K. Organic labyrinths and mazes[C] // *Proceedings of the 4th International Symposium on Non-Photorealistic Animation and Rendering*. New York: ACM Press, 2006: 79-86
- [16] Held M, Spielberg C. A smooth spiral tool path for high speed machining of 2D pockets[J]. *Computer-Aided Design*, 2009, 41(7): 539-550
- [17] Held M, Spielberg C. Improved spiral high-speed machining of multiply-connected pockets[J]. *Computer-Aided Design and Application*, 2014, 11(3): 346-357
- [18] Chiu W K, Yeung Y C, Yu K M. Tool-path generation for layer manufacturing of fractal objects[J]. *Rapid Prototyping Journal*, 2006, 12(4): 214-221
- [19] Piegl L, Tiller W. *The NURBS book*[M]. 2nd ed. New York: Springer Science & Business Media, 1997
- [20] Sarioz E. An optimization approach for fairing of ship hull forms[J]. *Ocean Engineering*, 2006, 33(16): 2105-2118