

基于非规则分块压缩的 3D 打印 稀疏矩阵存储与重构方法

徐敬华^{1,2)} 高铭宇²⁾ 苟华伟²⁾ 张树有^{1,2)} 谭建荣^{1,2)}

¹⁾(浙江大学浙江省先进制造技术重点实验室 杭州 310027)

²⁾(浙江大学机械工程学院 杭州 310027)

摘 要 现有 3D 打印(3D Printing, 3DP)通常是逐点伺服运动,成形效率低,使得技术正向高效高精度方向发展,例如数字光处理技术、选择性激光熔化、逐面打印等。通常,为了提高打印精度,需对层截面连通域进行更高分辨率栅格化,其后续光学转换等环节也因此生成更多元数据,导致切片非矢量化点阵数据量呈现大规模指数级增加,直接限制了打印件尺寸。为此,本文提出了一种基于非规则分块压缩(Irregular Block Compression, IBC)的 3D 打印稀疏矩阵存储与重构方法。首先,在初始模型坐标系构建沿坐标轴的 3D 凸包围盒(Axis-Aligned Bounding Boxes, AABB),得到流形网格模型的层截面多连通域,形成层截面掩模图,按照设定的分辨率生成栅格化点阵并转换成稀疏矩阵。根据稀疏度计算矩形规则块(Regular Block, RB)作为独立事件出现的概率化数学期望。结果表明,层截面矩阵数据的主要部分呈现非规则分块(Irregular Block, IB)特征,因此,压缩方法首要考虑非规则块的分布。进一步地,本文提出了稀疏矩阵非规则分块的概念,针对稀疏矩阵的非规则连通稀疏特征,将相邻行连通的非零块进行组合存储,构建互连通的非规则块,存储非零元素的数值及其有效的位置信息,获得首行索引、首列索引、连续数目及数值集进行无损压缩。按照非规则块进行层截面数据恢复与重构。通过计算相邻两层截面相似度,对 3D 实体模型进行多层连续面打印。以直列发动机缸体和多空格回转网环两种不同形态模型为例,与传统的压缩行存储(Compressed Row Storage, CRS)算法和分块压缩行存储(Block Compressed Row Storage, BCRS)算法相比,在存储量改进方面,IBC 方法比 CRS 改进可达 80.60%,比 BCRS 改进可达 14.62%,有效降低了算法的时间复杂度;在占用空间方面,IBC 方法比 BCRS 改进可达 22.56%,有效降低了算法的空间复杂度。IBC 方法特别适合层截面为区域化连通的 3D 打印稀疏矩阵的复杂形态模型的 3D 打印。

关键词 数字光处理技术;层截面掩模图;稀疏矩阵;概率化数学期望;非规则分块压缩;数据存储与重构

中图法分类号 TP312 **DOI 号** 10.11897/SP.J.1016.2020.02203

Storage and Reconstruction Method of Sparse Matrix for 3D Printing Based on Irregular Block Compression

XU Jing-Hua^{1,2)} GAO Ming-Yu²⁾ GOU Hua-Wei²⁾ ZHANG Shu-You^{1,2)} TAN Jian-Rong^{1,2)}

¹⁾(Key Lab of Advanced Manufacturing Technology of Zhejiang Province, Zhejiang University, Hangzhou 310027)

²⁾(College of Mechanical Engineering, Zhejiang University, Hangzhou 310027)

Abstract The current existing 3D Printing (3DP) is commonly characterized as servo motion point by point which reduces forming efficiency. So, it develops towards high efficiency and high precision, such as Digital Light Processing (DLP), Selective Laser Melting (SLM), section-by-section printing, etc. Generally, in order to improve the printing accuracy, it is necessary to rasterize the cross section connected domain with higher resolution whose subsequent optical conversion and other steps will generate more metadata at the same time. This in turn exponentially increases the

收稿日期:2018-06-04;在线发布日期:2019-11-29. 本课题得到国家自然科学基金(51935009, 51775494, 51821093)、浙江省科技计划研发项目(2019C01141, LGG20E050006)、国家重点研发计划(2018YFB1700701)资助。徐敬华, 博士, 副教授, 主要研究方向为 CS、CAGD。E-mail: xujh@zju.edu.cn。高铭宇, 硕士, 主要研究方向为 3D 打印。苟华伟, 硕士, 主要研究方向为 3D 打印。张树有(通信作者), 博士, 教授, 主要研究领域为 CAD。E-mail: zsy@zju.edu.cn。谭建荣, 博士, 教授, 主要研究领域为设计理论与方法。

data volume of slicing devectorization lattice to large scale, and limit the part size directly. Therefore, a storage and reconstruction method of sparse matrix for 3D printing based on Irregular Block Compression (IBC) is proposed. Firstly, in the original Model Coordinate System (MCS), the layered multi-connected domain is obtained via 3D Axis-Aligned Bounding Boxes (AABB) and the Layered Cross-sectional Mask (LCM) is generated. The rasterized lattice is constructed according to the preset resolution and converted to sparse matrix. The probabilistic mathematical expectation of rectangular Regular Blocks (RB) as independent events is calculated according to sparsity. The calculation proves that Irregular Blocks (IB) occupy major share in the sparse matrix. Therefore, the storage and reconstruction method should be optimized in primary consideration of the irregular blocks. Innovatively, the concept of irregular block of sparse matrix is proposed. Aiming at irregular connected sparse characteristics of sparse matrix, the connected nonzero block between adjacent rows are combinatorially stored to construct inter-connected irregular block. The value of non-zero elements and their valid location information is saved by lossless compression to obtain the first row index, the first column index, consecutive amount and value set. The data recovery and reconstruction of 3D printing layered section is carried out based on irregular blocks. The solid model is printed with multi-layer continuous surface by calculating the similarity of two adjacent sections. Taking in-line engine block and multi-genus porous revolution ring as examples, compared with traditional Compressed Row Storage algorithm (CRS) and Block Compressed Row Storage algorithm (BCRS), in terms of storage amount improvement, IBC is better than CRS with 80.60%, IBC is better than BCRS with 14.62%, which effectively reduces the time complexity of the algorithm. In terms of storage space, IBC is better than BCRS with 22.56%, which effectively reduces the space complexity of the algorithm. The IBC is especially useful for 3D printing of complex shape models whose layered section is region connected large-scale sparse matrix.

Keywords Digital Light Processing (DLP); Layered Cross-Sectional mask (LCM); sparse matrix; probabilistic mathematical expectation; Irregular Block Compression (IBC); data storage and reconstruction

1 引 言

增材制造 (Additive Manufacturing, AM) 通过增加材料来造物^[1], 能够直接从三维模型数据生成终端实物. 目前已有的 3D 打印 (3D Printing, 3DP) 成形工艺方法^[2-5] 通常是控制喷头或激光沿截面轮廓由点到线、由线到面的成形, 效率低, 难以满足高品质高效生产的需求^[6-7]. Nature 期刊刊登了高强度铝合金的 3D 打印方法^[8] 及复合材料的高能效快速制造应用^[9]. Science 期刊刊登了美国 Carbon 3D 公司的 Tumbleston 等人^[10] 在光固化打印的基础上提出的连续液面制造技术, 即将立体光固化成型法 (Stereo Lithography Appearance, SLA) 的逐点固化改为逐面固化, 提升了打印速度. 美国 Lawrence Livermore 国家实验室的 Matthews 等人^[11] 2017 年提出了光寻址光阀技术 (Optically Addressable Light

Valves, OALV), 可将入射激光转化成多激光束阵列面, 实现了金属的面打印, 该技术打印出的小型涡轮叶片直径仅为 $\phi 13$ mm, 这种面打印的技术使生产效率提升, 但打印尺寸受局限, 还有待进一步研究. 美国 UC 伯克利和 Lawrence Livermore 国家实验室的 Kelly 等人^[12] 把一组二维图像, 从不同的角度投射并曝光叠加, 让光敏液体 (甲基丙烯酸酯明胶水凝胶) 固化成所需要的几何形状, 打印只需几十秒, 最高精度可达 0.3 mm. 近年来涌现的数字光处理技术 (Digital Light Processing, DLP), 其专利核心是数字微镜元件 (Digital Micromirror Device, DMD). Mu 等人^[13] 通过添加多壁碳纳米管的混合物制造了导电复合结构, 如电激活的形状记忆复合结构, 为 4D 打印提供了新的技术支持. 德克萨斯大学卫生科学中心 Kadry 等人^[14] 利用 DLP 制造了口服片状药物的载体平台, 载体平台释放特性可根据需求进行调整. He 等人^[15] 利用 DLP 技术制造了带有退刀槽

和蜂窝状复杂结构的氧化铝材质的切削工具, 实现了陶瓷材料的面打印。

随着成形件尺寸增大及打印分辨率增加, 其后续光学转换等环节也生成更多元数据, 导致非矢量化点阵数据量指数级增加。稀疏矩阵中的零元素占比非常大, 需要大量存储空间, 很多稀疏矩阵的优化算法被提出, 例如锯齿状对角线存储 (Jagged Diagonal Storage, JDS) 算法、天际线存储 (Skyline storage, SKY) 算法、缓存遗忘扩展二叉树存储 (Cache Oblivious Extension Quadtree Storage, COEQT) 算法、压缩行存储 (Compressed Row Storage, CRS) 算法、分块压缩行存储 (Block Compressed Row Storage, BCRS) 算法、压缩对角线存储 (Compressed Diagonal Storage, CDS) 算法等^[16]。

美国 Sandia 国家实验室的 Hoemmen 等人^[17] 在 BCRS 的基础上, 优化了稀疏矩阵向量乘法, 减少了带宽占用, 提高了处理速度。德国开姆尼茨工业大学的 Richter 等人^[18] 优化了常用的三种稀疏矩阵格式, 节省了排序以及动态数据结构所占用的存储空间。美国乔治亚理工学院的 Liu 等人^[19] 在 CRS 的基础上提出了一种新型的兼容统一计算设备架构的稀疏矩阵向量乘法, 提高了处理速度。印度理工学院的 Singh 等人^[20] 基于 BCRS 提出了一种新型的储存格式, 优化了存储效率, 并在图形处理单元加速的系统上实现了稀疏矩阵向量乘法的自动调整。中佛罗里达大学 Montagne 等人^[21] 基于 JDS 提出了一种转置锯齿状对角线存储 (Transpose Jagged Diagonal Storage, TJDS) 算法, 这种算法适用于更多稀疏模式的矩阵, 而且所需的线型数组比 JDS 的四个数组少一组, 所以其压缩率更高。佐治亚理工学院 Belgin 等人^[22] 提出了新型的基于模式表示 (Pattern-Based Representation, PBR) 算法, 利用重复的非零块储存, 并以块坐标的格式表示所有由该非零块模式组成的子矩阵, 从而减少了重复单元所占用的储存空间。德国埃朗根-纽伦堡大学 Kreutzer 等人^[23] 提出了一种填充的锯齿状对角线存储 (Padded Jagged Diagonals Storage, PJDS) 算法, 在通用计算图形处理器上的内存占用减少多达 70%。波兰弗罗茨瓦夫大学 Koza 等人^[24] 提出了一种用于稀疏矩阵存储的压缩多行存储格式, 包含一个数组和三个辅助整数数组, 能够快速地完成转换。Chalauri 等人^[25] 发明了储存非零数值及其列索引的锯齿状非零子矩阵数据结构 (Jagged Non-Zero sub-matrix data structure, JNZ), 可以用于矩阵的复杂计算。Xu 等人^[26] 将传统的 BCRS 算法与哈夫曼编码相结合, 改进了稀疏矩

阵的存储结构, 减少了数据复杂度。He 等人^[27] 通过一阶函数计算二维信号的稀疏度, 并利用复杂度较低的二维梯度投影方法解决了稀疏矩阵的恢复问题。Chen 等人^[28] 基于 CSR, 利用单指令多数据流 (Single Instruction Multiple Data, SIMD) 提出了 CRS-SIMD 存储格式, 新的存储格式可以通过矢量化来提高存储效率。

本文提出一种非规则分块压缩 (Irregular Block Compression, IBC) 的 3D 打印稀疏矩阵存储与重构方法, 考虑层截面矩阵数据的稀疏性与相似性^[29-30], 对数据进行大幅度压缩, 降低数据的存储量 (存储矩阵所需的整型数的数目), 能够更好地适应面打印的对于高精度、高分辨率、大尺寸的打印需求。

2 3D 打印流形模型层截面掩模图栅格化点阵构建

2.1 3D 打印几何模型层截面掩模图及其栅格化

目前 3D 打印的流形几何模型切片通常是基于 STL 模型进行的, 从而获取层截面的掩模图。对于模型的不同精度要求, 通常也会有不同的分层厚度。通过分层厚度可以计算出总的分层层数, 每层建立一个用于存储有向多边形数据结构。遍历 STL 模型中所有的三角片, 逐层求取每个分层的轮廓。轮廓是由一系列有序的点集合并按顺序连接构成的 1 个或多个多边形。其中多边形的边界具有方向性, 沿正方向移动时多边形内部区域位于行进方向左侧。

在初始模型坐标系 (Model Coordinate System, MCS), 求解三维模型的沿坐标轴的凸包围盒 (Axis-Aligned Bounding Boxes, AABB), 沿 x, y, z 轴的边长记为 x_b, y_b, z_b 。通过分层切片获得不同归一化高度比例 h_i 的层截面多连通域。

$$h_i = \frac{z_i}{z_b} \times 100\%, h_i \in [0, 1] \quad (1)$$

式中, z_i 为第 i 层的高度, z_b 为 AABB 的 z 行程 Δz 。

将每一层的截交线构成的复合多边形轮廓, 转换为多连通域, 形成层截面掩模图 (Layered Cross-sectional Mask, LCM)。栅格化是指对二维 LCM 生成最小的矩形包围盒, 并按某一长度划分为单元格的过程。按分辨率 r_x, r_y 要求确定在 x, y 方向上的二维栅格边长 l_m, l_n , 其中 m, n 分别为所需划分的栅格数目, 则

$$m = \text{ceil}\left(\frac{x_{\max} - x_{\min}}{l_m}\right) \leq r_x \quad (2)$$

$$n = \text{ceil}\left(\frac{y_{\max} - y_{\min}}{l_n}\right) \leq r_y \quad (3)$$

式中,ceil 表示向上取整,总栅格数为 $m \times n$.

通常, l_m, l_n 对于光固化成型 (Stereo Lithography Appearance, SLA) 工艺可取 $0.05 \sim 0.025 \text{ mm}$, DLP 可取 $0.025 \sim 0.035 \text{ mm}$, SLM 可取 $0.02 \sim 0.05 \text{ mm}$.

以图 1 的 V8 发动机缸体 LCM 为例,若尺寸或分辨率仅增加 1 倍, $m \times n$ 从 100×100 增长到 200×200 ,则需要被存储的栅格数由 2903 增长到 11 854,增长了 308.34%,可见,数据量的指数级增加,限制了打印件尺寸.

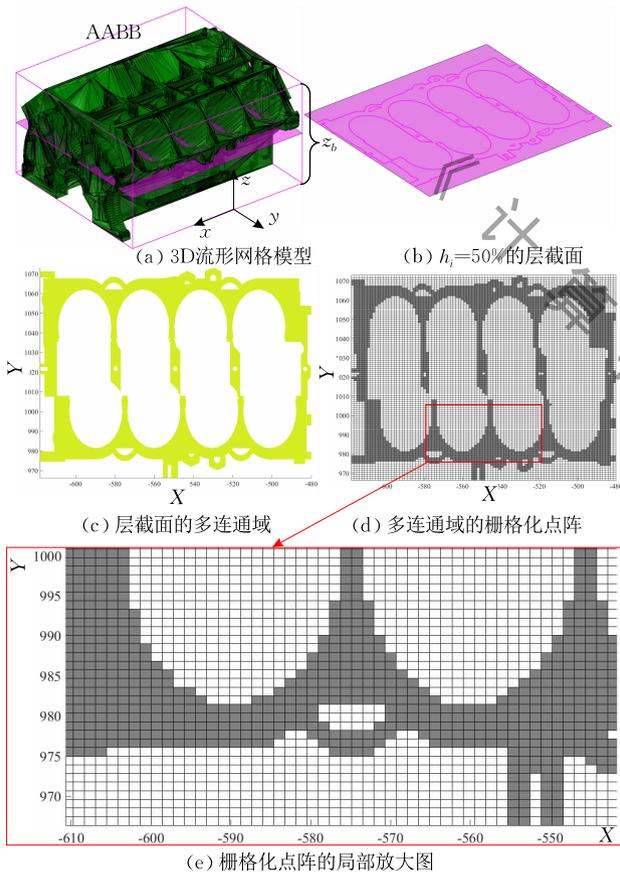


图 1 3D 流形网格模型及层截面连通域 LCM

2.2 稀疏矩阵的规则块与非规则块

稀疏矩阵中一般会存在大量的非零块,包括规则块 (Regular Block, RB) 和非规则块 (Irregular Block, IB). 图 2 所示,为一个简化稀疏矩阵,其中含

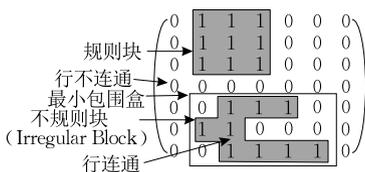


图 2 稀疏矩阵中具有连通特征规则块和非规则块

有一个每行非零元素数目不完全相同的非规则块和一个每行非零元素数目完全相同的矩形规则块.非规则块内部和矩形规则块内部都是行连通的,而它们之间是不连通的.传统的 BCRS 算法将稀疏矩阵分为若干个矩形规则非零块,对于非规则块则以最小包围盒的形式进行分块,这会记录一些零元素.而 IBC 算法则直接记录非规则块,从而减少储存量,提高处理速度.

稀疏度指稀疏矩阵中空(零)元素在总单元中所占的比例.将稀疏度记为 s .在一个行数为 m ,列数为 n 的稀疏矩阵中

$$s = \frac{n_0}{m \times n} \times 100\%, s \in [0, 1] \quad (4)$$

式中, n_0 为矩阵中数值为零的元素数目.

在 BCRS 算法中储存矩形规则非零块所需数据存储量为 4,具体为矩形规则非零块第一行的行数、第一列的列数、行数和列数.

如果矩形规则非零块的边长小于等于 2,则 BCRS 算法压缩率为 1,所以大于 3×3 的矩形规则非零块数目直接影响 BCRS 算法的压缩效率,下面对其数目进行概率分析.

可能存在 3×3 矩形规则非零块的位置数目的期望为 $E_{3 \times 3} = (m-2) \times (n-2)$,对于 3×4 的非零块,存在的可能位置数期望 $E_{3 \times 4} = (m-2) \times (n-3)$,由归纳法,对于单个 $k \times l$ 的非零块,存在的可能位置数期望有

$$E_{k \times l} = (m-k+1) \times (n-l+1) \quad (5)$$

由此,可推出 3×3 的矩形规则非零块数目的最大期望如下:

$$\begin{aligned} M_{3 \times 3} &= E_{3 \times 3} \times (1-s)^{3 \times 3} \times s^{(3+3) \times 2} - \\ &\quad o((1-s)^{3 \times 3} \times s^{(3+3) \times 2}) \\ &\approx E_{3 \times 3} \times (1-s)^{3 \times 3} \times s^{(3+3) \times 2} \end{aligned} \quad (6)$$

$k \times l$ 的矩形规则非零块数目最大期望:

$$M_{k \times l} = E_{k \times l} \times (1-s)^{k \times l} \times s^{(k+l) \times 2} \quad (7)$$

所以总最大期望:

$$M = \sum_{i=3}^m \sum_{j=3}^n M_{i \times j} \quad (8)$$

将稀疏矩阵大小及其稀疏度代入式(8)可以计算出矩形规则非零块数目的最大期望.如图 3(a)所示, 1000×1000 的稀疏矩阵中的矩形规则非零块数目在 $s=57\%$ 取到最大期望值为 0.6202.如图 3(b)所示, $s=57\%$ 时在 $m=1500, n=1500$ 处矩形规则非零块数目取到最大期望值为 1.397.因此在维度不超过 1500×1500 、任意稀疏度的矩阵中,出现

3×3 以上的矩形规则非零块的最大数量的数学期望甚至不足 2, 由补集理论可知非规则非零块的数量远大于矩形规则非零块的数量, 进一步说明: 传统的 BCRS 存储算法对于此类数据压缩率低. 矩形规则非零块越少, 则 IBC 算法相对于 BCRS 算法越有优势.

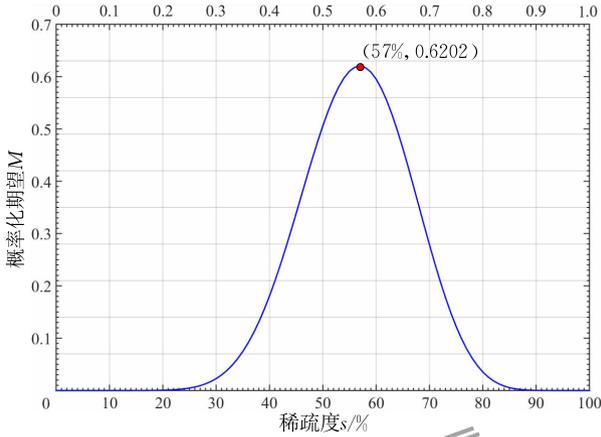
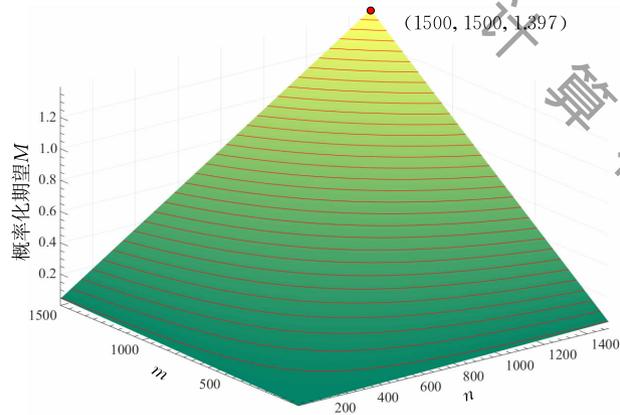
(a) $m=1000, n=1000$ (b) $s=57\%$

图 3 矩形非零块的概率化数学期望

3 行连通的非规则分块压缩算法与流程

3.1 行连通的非规则分块压缩算法

由 2.2 节可以得到, 矩形规则非零块的期望值 $M < 2$, 稀疏矩阵中的主要成分为非规则非零块. 所以针对非规则非零块对 BCRS 算法进行改进. 将原始的稀疏矩阵划分为多个内部行连通的非规则非零块进行组合储存. 从而避免零元素占用存储空间, 保证存储空间充分利用, 减少占用的存储量. $m \times n$ 的稀疏矩阵如矩阵式(9).

$$\mathbf{A} = \begin{bmatrix} 0 & \cdots & 0 & x_{1,1} & \cdots & x_{1,n_1} & 0 & \cdots & 0 \\ 0 & x_{2,1} & \cdots & x_{2,n_2} & 0 & & \cdots & & 0 \\ \vdots & & & & \ddots & & & & \vdots \\ x_{i,1} & \cdots & x_{i,n_i-1} & 0 & & \cdots & & 0 & x_{i,n_i} \\ \vdots & & & & \ddots & & & & \vdots \\ x_{m,1} & \cdots & x_{m,k} & 0 & \cdots & 0 & x_{m,k+1} & \cdots & x_{m,m_n} \end{bmatrix} \quad (9)$$

区别于 BCRS 算法, IBC 算法不填充零元素, 而是记录非规则非零块的位置及形状, 并直接储存非规则非零块.

IBC 算法将稀疏矩阵转换为四个数组进行存储, 其中:

\mathbf{V} 中记录了所有非零元素的值; \mathbf{C} 中记录了非零块每行的首列索引; \mathbf{R} 中记录了非零块每行非零元素的数目; \mathbf{B} 中记录了非零块的首行索引. 四个数组均以非零块序号为第一顺序, 以行数为第二顺序进行由小到大的有序记录.

将如矩阵式(9)所示的稀疏矩阵 \mathbf{A} 分为两个非规则非零块, 其存储算法结构如下:

$$\mathbf{V} = \begin{bmatrix} x_{1,1}, \cdots, x_{1,n_1}, x_{2,1}, \cdots, x_{2,n_2}, \cdots, x_{i,1}, \cdots, x_{i,n_i-1}, \cdots, x_{m,1}, \cdots, x_{m,k} \\ x_{i,n_i}, \cdots, x_{m,k}, \cdots, x_{m,n_m} \end{bmatrix} \quad (10)$$

$$\mathbf{C} = \begin{bmatrix} \text{col}(x_{1,1}), \text{col}(x_{2,1}), \cdots, \text{col}(x_{i,1}), \cdots, \text{col}(x_{m,1}) \\ \text{col}(x_{i,n_i}), \cdots, \text{col}(x_{m,n_m-k+1}) \end{bmatrix} \quad (11)$$

$$\mathbf{R} = \begin{pmatrix} n_1, n_1 + n_2, \cdots, n_1 + n_2 + \cdots + n_i - 1, \cdots, n_1 + n_2 + \cdots + k \\ 1, \cdots, 1 + \cdots + n_m - k + 1 \end{pmatrix} \quad (12)$$

$$\mathbf{B} = (1, i) \quad (13)$$

进一步推广到一般情况:

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 \\ \vdots \\ \mathbf{V}_q \end{bmatrix} \quad (14)$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_1 \\ \vdots \\ \mathbf{C}_q \end{bmatrix} \quad (15)$$

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 \\ \vdots \\ \mathbf{R}_q \end{bmatrix} \quad (16)$$

$$\mathbf{B} = (b_1, \cdots, b_q) \quad (17)$$

$$\mathbf{A}_s = f_1(\mathbf{A}, m, n, i, j) \quad (18)$$

其中 \mathbf{A} 表示原始稀疏矩阵, \mathbf{A}_s 表示压缩后的四个数组, m, n 分别表示 \mathbf{A} 的行数和列数, i, j 表示非零元

素的行索引和列索引,令非零元素的数目为 v ,非规则非零块的数目为 q ,非规则非零块的行数总和数目为 a ,则存储总量为 $v+q+2a$.

$$F: \begin{pmatrix} \mathbf{V} \rightarrow v \\ \mathbf{C} \rightarrow a \\ \mathbf{R} \rightarrow a \\ \mathbf{B} \rightarrow q \end{pmatrix} \quad (19)$$

一个 $m \times n$ 的仅含有 0 和 1 两种元素的稀疏矩阵,其每行非零元素的数目分别为 a_1, a_2, \dots, a_m ,元素总的数目为 $\mathbf{V}(m) = a_1 + a_2 + \dots + a_m$,在 BCRS 算法中每行非零块数目分别为 b_1, b_2, \dots, b_m ,总的数目为 $\mathbf{B}(m) = b_1 + b_2 + \dots + b_m$.按照 CRS 的算法结构,可算出存储总量 T_{CRS} .

$$T_{\text{CRS}} = \mathbf{V}(m) + m \quad (20)$$

按照 BCRS 算法结构,算出存储总量 T_{BCRS} .

$$T_{\text{BCRS}} = \mathbf{B}(m) + \mathbf{B}(m) + m = 2\mathbf{B}(m) + m \quad (21)$$

按照 IBC 算法结构,非规则块的数目为 r ,每个非规则块的行数分别为 c_1, c_2, \dots, c_m ,记 $\mathbf{C}(r) = c_1 + c_2 + \dots + c_m$.算出存储总量 T_{IBC} .

$$T_{\text{IBC}} = \mathbf{C}(r) + \mathbf{C}(r) + r = 2\mathbf{C}(r) + r \quad (22)$$

因为大部分非零块所含元素数目都大于 2,所以在稀疏矩阵中, $\mathbf{V}(m)$ 远大于 $\mathbf{B}(m)$,而 $\mathbf{B}(m)$ 近似于 $\mathbf{C}(r)$.同时由于非规则非零块的行数一般大于 2,所以 m 远大于 r .可以得出 $T_{\text{IBC}} < T_{\text{BCRS}} < T_{\text{CRS}}$.可见,相比于 BCRS 算法和 CRS 算法,在存储总量方面 IBC 算法更具有优势.

3.2 非规则分块压缩算法流程

传统的分块压缩行存储(Block Compressed Row Storage,BCRS)算法在非规则非零块的数量远大于矩形规则非零块的数量时,对于此类数据压缩率低.非规则分块压缩(Irregular Block Compression,IBC)算法的设计动机就是为解决此问题,具体步骤为已知输入稀疏矩阵 \mathbf{A} ,获得 \mathbf{A} 中非零块的首行索引 \mathbf{B} ; \mathbf{A} 中非零块每行的首列索引 \mathbf{C} ; \mathbf{A} 中非零块每行非零元素的数目 \mathbf{R} ; $\mathbf{B}, \mathbf{C}, \mathbf{R}$ 总的存储量 S .

算法 1. IBC 算法.

输入: \mathbf{A}

输出: $\mathbf{B}, \mathbf{C}, \mathbf{R}, S$

//初始化中间变量 $i, j, k, p_1, m_1, m_2, g, r, c, b$ 和中间矩阵 $\mathbf{q}, \mathbf{X}, \mathbf{X}_1$.

1. FOR $i=1$ TO m DO
2. FOR $j=1$ TO n DO
3. IF $\mathbf{A}(i, j) > 0$ THEN

4. $k=k+1$; $\mathbf{q}(k, 1)=i$; $\mathbf{q}(k, 2)=j$; %用三元组法表征稀疏矩阵
5. $p_1 = \text{rows}(\mathbf{q})$; % p_1 为 \mathbf{A} 中非零元素的数目
6. $m_1 = p_1$;
7. FOR $i=1$ TO p_1 DO %每次循环确定一个非规则非零块的列索引和数目索引,最多循环 p_1 次
8. $g = \mathbf{q}(1, 1) - 1$; $\mathbf{X} = []$; $r = 0$; $\mathbf{B} = [\mathbf{B}, \mathbf{q}(1, 1)]$; % g 表示上一个非零元素的行数, \mathbf{X} 矩阵记录第 i 个非零块所包含的元素序号, \mathbf{B} 矩阵记录第 i 个非零块的第一行行数
9. FOR $j=1$ TO m_1 DO % m_1 为第 i 次循环时剩余元素的数目
10. IF $\mathbf{q}(j, 1) - g = 1$ THEN %当相邻元素位于连续行时
11. $c = \mathbf{q}(j, 2)$; $g = \mathbf{q}(j, 1)$; $b = j$; $\mathbf{C} = [\mathbf{C}, c]$; % \mathbf{C} 矩阵记录当前行首个非零元素的列数
12. $[\mathbf{X}] = [\mathbf{X}, j]$; $\mathbf{RR} = [\mathbf{RR}, r]$; $r = r + 1$; % \mathbf{RR} 记录当前非零块当前行非零元素的累计数目
13. ELSE IF $\mathbf{q}(j, 2) - \mathbf{q}(b, 2) = j - b$ && $\mathbf{q}(j, 1) = \mathbf{q}(b, 1)$ THEN %当相邻元素位于同行相邻列时
14. $\mathbf{X} = [\mathbf{X}, j]$; $r = r + 1$;
15. $\mathbf{R} = [\mathbf{R}, \mathbf{RR}]$; $m_2 = \text{length}(\mathbf{X})$; $\mathbf{X}_1 = \text{fliplr}(\mathbf{X})$; %将每个非零块中各行非零元素的累计数目索引记录至 \mathbf{R} , m_2 记录已被压缩的元素数目,将 \mathbf{X} 左右翻转,即将所含序号从大到小排列
16. FOR $k=1$ TO m_2 DO
17. $\mathbf{q}(\mathbf{X}_1(k), :) = 0$; %将已经压缩的元素删除
18. $m_1 = \text{rows}(\mathbf{q})$; $\mathbf{q} = \mathbf{q}$; %更新 m_1 为当前剩余元素数目
19. IF $m_1 = 0$ THEN %当所有元素均被压缩后结束
20. BREAK;
21. $S = \text{length}(\mathbf{C}) + \text{length}(\mathbf{R}) + \text{length}(\mathbf{B})$; %统计总存储量
22. RETURN $\mathbf{B}, \mathbf{C}, \mathbf{R}, S$

4 压缩数据重构与层间差分存储

4.1 非规则分块压缩算法的数据重构

在面打印时,根据压缩所得的四个数组进行数据重构,得到原始稀疏矩阵,从而控制激光阵列.式(10)~(18)为矩阵式(9)所示的稀疏矩阵 \mathbf{A} 通过 IBC 算法压缩存储的结果,根据 IBC 存储原理进行反推可得原始稀疏矩阵 \mathbf{A} ,如式(23)所示.

$$\mathbf{A}_t = f_2(\mathbf{V}, \mathbf{B}, \mathbf{C}, \mathbf{R}) \quad (23)$$

IBC 算法的重构首先需要读取数组 \mathbf{B} 的元素数目,其代表了非规则非零块的数目,对应 $\mathbf{V}, \mathbf{B}, \mathbf{C}$ 数组的不同部分,利用数据恢复函数,即可得出每一非

规则块的数据信息,即

$$\begin{aligned} \mathbf{A}_1 &= f_2(V_1, B_1, C_1, R_1) \\ \mathbf{A}_2 &= f_2(V_2, B_2, C_2, R_2) \\ &\dots \\ \mathbf{A}_q &= f_2(V_q, B_q, C_q, R_q) \end{aligned} \quad (24)$$

得到所有非规则非零块的位置信息和形状信息,并将其合并得到原稀疏矩阵 \mathbf{A} . 数据重构是按照非规则非零块进行数据还原,并包含了非零块的全部信息,是无损压缩.

$$\mathbf{A} = \mathbf{A}_1 + \mathbf{A}_2 + \dots + \mathbf{A}_q \quad (25)$$

4.2 层截面掩模图的差分存储

通过减少 IBC 算法对原始几何模型层截面掩模图进行数据压缩,可以减少存储量. 对于 3D 打印流形模型,通过同一包围盒构建栅格化点阵,可获得多个大小相同的层截面矩阵 \mathbf{A} , 仅需差运算,即可得到差矩阵 $\Delta\mathbf{A}$, 并求出相邻两层的相似度 κ , 以决定该层的最简数据存储方式.

层截面矩阵集合 \mathbf{A} :

$$\begin{aligned} \mathbf{A} &= \{\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3, \dots, \mathbf{A}_n\} \\ \mathbf{A}_i &\text{为 } m \times n \text{ 的矩阵 } \& i \in [1, n] \end{aligned} \quad (26)$$

层截面差矩阵序列集合:

$$\begin{aligned} \Delta\mathbf{A} &= \text{diff}(\mathbf{A}) = \{\Delta\mathbf{A}_1, \Delta\mathbf{A}_2, \Delta\mathbf{A}_3, \dots, \Delta\mathbf{A}_{n-1}\} \\ \Delta\mathbf{A}_i &\in R^{m \times n} \& i \in [1, n-1] \end{aligned} \quad (27)$$

式中, diff 为 1 阶前向差分, $\forall \Delta\mathbf{A}_i = \mathbf{A}_{i+1} - \mathbf{A}_i$, $i \in [1, n-1]$.

第 i 层与第 $i+1$ 层之间的相似度 κ_i 可定义为

$$\kappa_i = 1 - \frac{m \times n - n_0(\Delta\mathbf{A}_i)}{m \times n - n_0(\mathbf{A}_i)}, \quad i \in [1, n-1] \quad (28)$$

层截面掩模图最简数据的差分存储流程如下:

Step1. $i=1$, 将第 i 层的 LCM 转化为矩阵, 并储存到集合 \mathbf{A} ;

Step2. 根据打印精度需求 r_x, r_y , 确定阈值 δ_i ;

Step3. $i=i+1$, 判断第 i 层的掩模图矩阵与第 $i-1$ 层的矩阵相似度 κ_{i-1} 是否大于阈值 δ_i , 若是, 则转入 Step4, 否则转入 Step5;

Step4. 计算第 i 层与第 $i-1$ 层的差矩阵 $\Delta\mathbf{A}_{i-1}$, 并储存到集合 $\Delta\mathbf{A}$, 转入 Step6;

Step5. 计算第 i 层矩阵, 并储存到集合 \mathbf{A} , 转入 Step6;

Step6. 判断 i 是否等于层截面数目, 若是, 则结束流程, 否则转入 Step2.

获得层截面矩阵集合 \mathbf{A} 及层截面差矩阵序列集合 $\Delta\mathbf{A}$.

5 实例

5.1 直列四缸缸体实例

如图 4 所示的 L4 直列四缸缸体虚拟打印模型, 大于平均曲率的顶点有 57 021 个, 占 33.36%, 其 AABB 的边长比 $x_b : y_b : z_b = 1.2500 : 1.6723 : 1$.

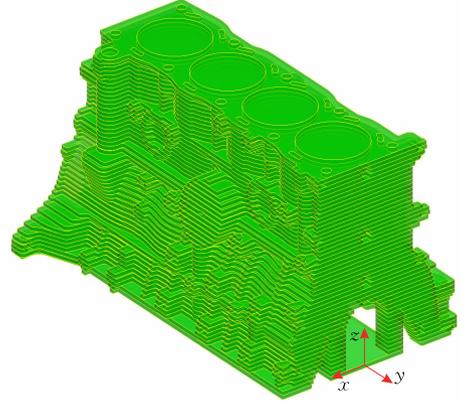


图 4 L4 发动机缸体虚拟打印模型

如图 5 所示, 为每层掩模图 LCM 的稀疏度, 最大值 $\max(s) = 82.8871\%$, $h_i = 86\%$, 最小值 $\min(s) = 48.7718\%$, $h_i = 2\%$, 平均值 $\text{mean}(s) = 78.5616\%$.

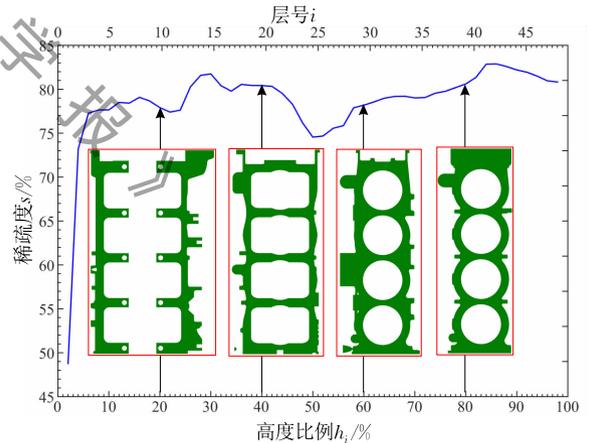


图 5 L4 发动机缸体 LCM 的稀疏矩阵稀疏度

如图 6 所示, 为 L4 缸体模型的切片栅格化点阵, 稀疏度分别为 78.66%、80.42%、78.19%、80.6%.

取图 6(a) 的 0.2 高度切片栅格, 按照 BCRC 和 IBC 两种方法进行计算(红线与蓝线绘图, 下同), 可以分别得出 \mathbf{C} 、 \mathbf{R} 、 \mathbf{B} 的数据信息. 存储算法结构中 \mathbf{C} 、 \mathbf{R} 、 \mathbf{B} 的存储数据图, 如图 7 所示. 存储量的改进方面, 主要表现在 \mathbf{B} 矩阵上, 其存储量由 BCRC 的 77(图 7(c))改进到 IBC 的 10(图 7(f)), 改进了 87.01%, 降

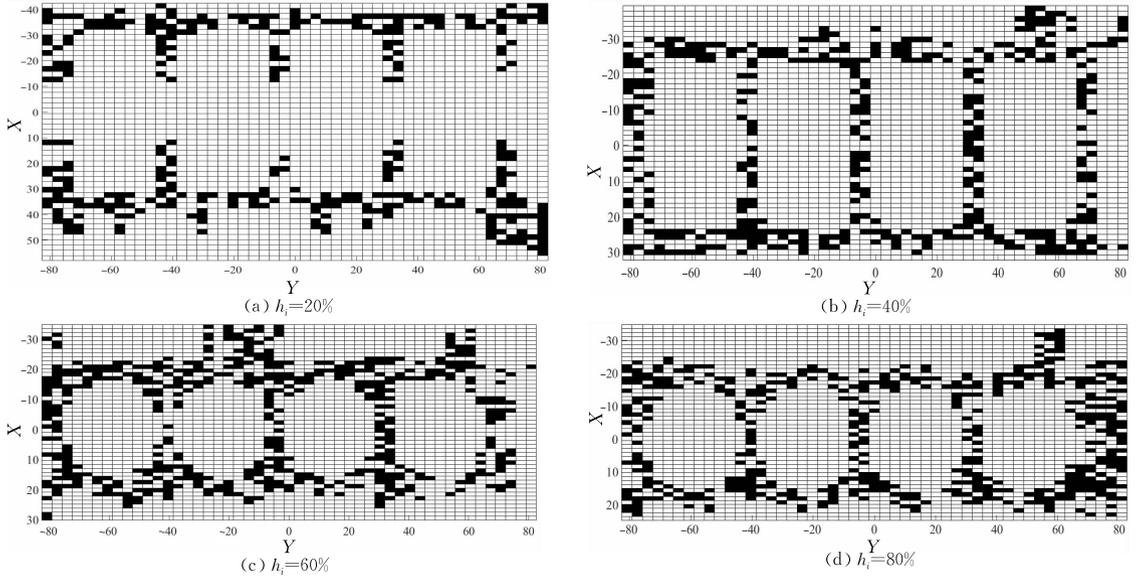


图 6 L4 缸体模型的切片栅格化点阵

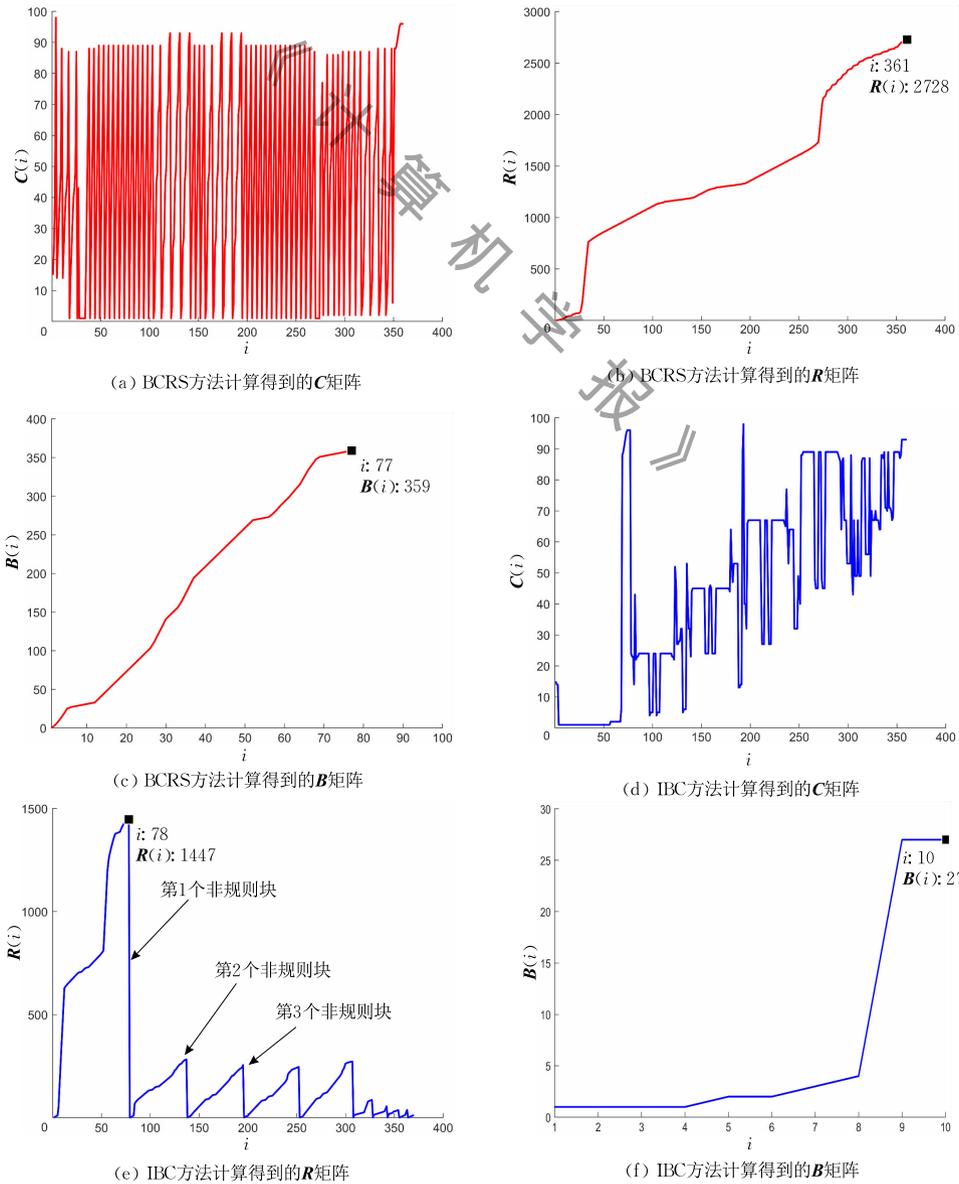


图 7 L4 缸体模型的 BCRS 算法与 IBC 的对比图

低了算法的时间复杂度。同时,占用空间的改进方面,主要表现在 R 矩阵上,其所储存的最大值由 BCRS 的 2728(图 7(b))改进到 IBC 的 1447(图 7(e)),改进了 46.96%,降低了算法的空间复杂度。

表 1 为发动机缸体 4 个层截面(图 6)根据不同算法压缩的对比。可以看出,在存储量改进方面,IBC 算法比 CRS 节省,最高为 80.60%,最低为 73.62%,平均为 76.34%;比 BCRS 算法节省,最高为 9.46%,最低为 7.3%,平均为 8.39%,降低了算法的时间复杂度。在占用空间改进方面(其中,字节数 Byte=1KB/1024),IBC 算法比 BCRS 算法节省,最高为 21.57%,最低为 14.12%,平均为 17.11%,降低了算法的空间复杂度。

表 1 L4 缸体模型的不同算法的比较

h_i	存储量与字节数					
		CRS	BCRS	IBC	IBC 比 CRS 节省/%	IBC 比 BCRS 节省/%
20%	存储量	2805	798	740	73.62	7.30
	字节数	674	1133	973		14.12
40%	存储量	3423	970	886	74.12	8.66
	字节数	764	1303	1022		21.57
60%	存储量	4144	888	804	80.60	9.46
	字节数	817	1290	1084		15.97
80%	存储量	4037	1010	928	77.01	8.12
	字节数	879	1456	1212		16.76

5.2 多亏格回转网环实例

多亏格网环虚拟打印如图 8 所示,形态多变,欧拉示性数为 -648,亏格(孔洞)数为 325,其 AABB 的边长比 $x_b:y_b:z_b=2.8470:2.8487:1$ 。

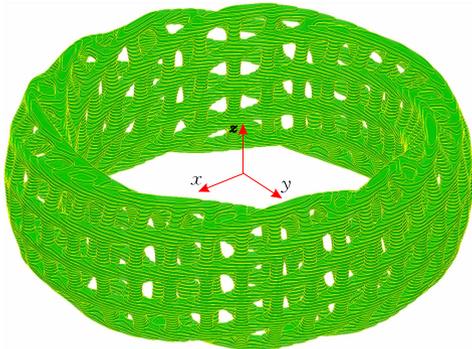


图 8 多亏格网环虚拟打印模型

如图 9 所示,为每层掩模图 LCM 的稀疏度,最大值 $\max(s)=97.6435\%$, $h_i=98\%$,最小值 $\min(s)=88.45\%$, $h_i=12\%$,平均值 $\text{mean}(s)=89.8429\%$ 。

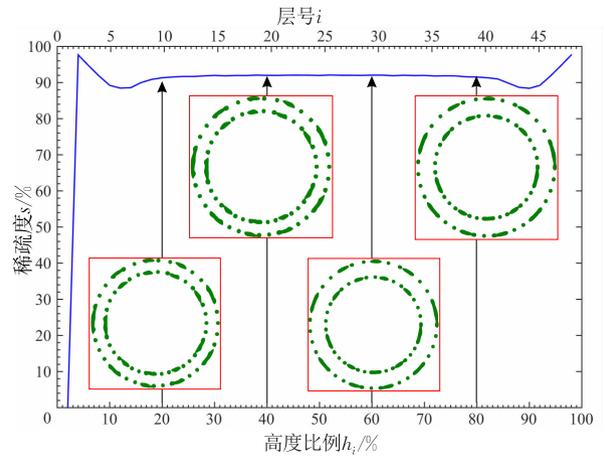


图 9 多亏格网环的 LCM 的稀疏矩阵稀疏度

如图 10 所示,(a)、(b)、(c)、(d)为多亏格网环模型的切片栅格图,稀疏度分别为 90.87%、91.96%、91.99%、91.59%。

取图 10(a)的 0.2 高度切片栅格图,按照 IBC 和 BCRC 两种算法进行计算,可以分别得出 C 、 R 、 B 的数据信息。存储算法结构中 C 、 R 、 B 的存储数据图,如图 11 所示。存储量的改进方面,主要表现在 B 矩阵上,其存储量由 BCRC 的 196(图 11(c))改进到 IBC 的 6(图 11(f)),改进了 96.94%,降低了算法的时间复杂度。同时,占用空间的改进方面,主要表现在 R 矩阵上,其最大值由 BCRC 的 3559(图 11(b))改进到 IBC 的 1264(图 11(e)),改进了 64.48%,降低了算法的空间复杂度。

表 2 为图 10 多亏格网环的 4 个层截面使用不同压缩算法进行压缩的效果对比。根据表 2,可以看出对于图 10 的 4 组层截面,在存储量改进方面,IBC 算法比 CRS 算法节省,最高为 69.85%,最低为 67.77%,平均为 68.83%;比 BCRC 算法节省,最高

表 2 多亏格网环模型的不同算法的比较

h_i	存储量与字节数					
		CRS	BCRS	IBC	IBC 比 CRS 节省/%	IBC 比 BCRS 节省/%
20%	存储量	3755	1317	1132	69.85	14.05
	字节数	1702	2163	1096		21.27
40%	存储量	3420	1282	1096	67.95	14.51
	字节数	1681	2105	1646		21.81
60%	存储量	3370	1272	1086	67.77	14.62
	字节数	1642	2083	1613		22.56
80%	存储量	3763	1322	1138	69.76	13.92
	字节数	1708	2157	1710		20.72

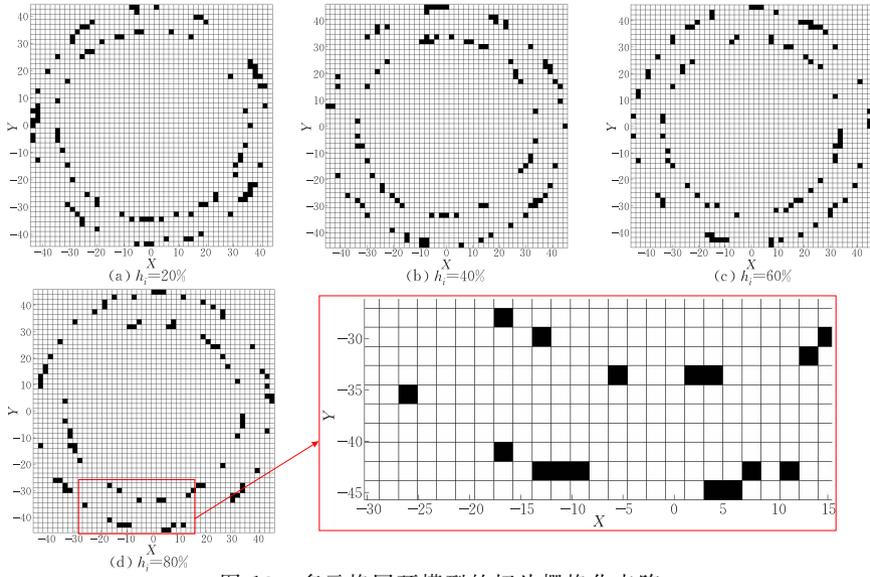


图 10 多亏格网环模型的切片栅格化点阵

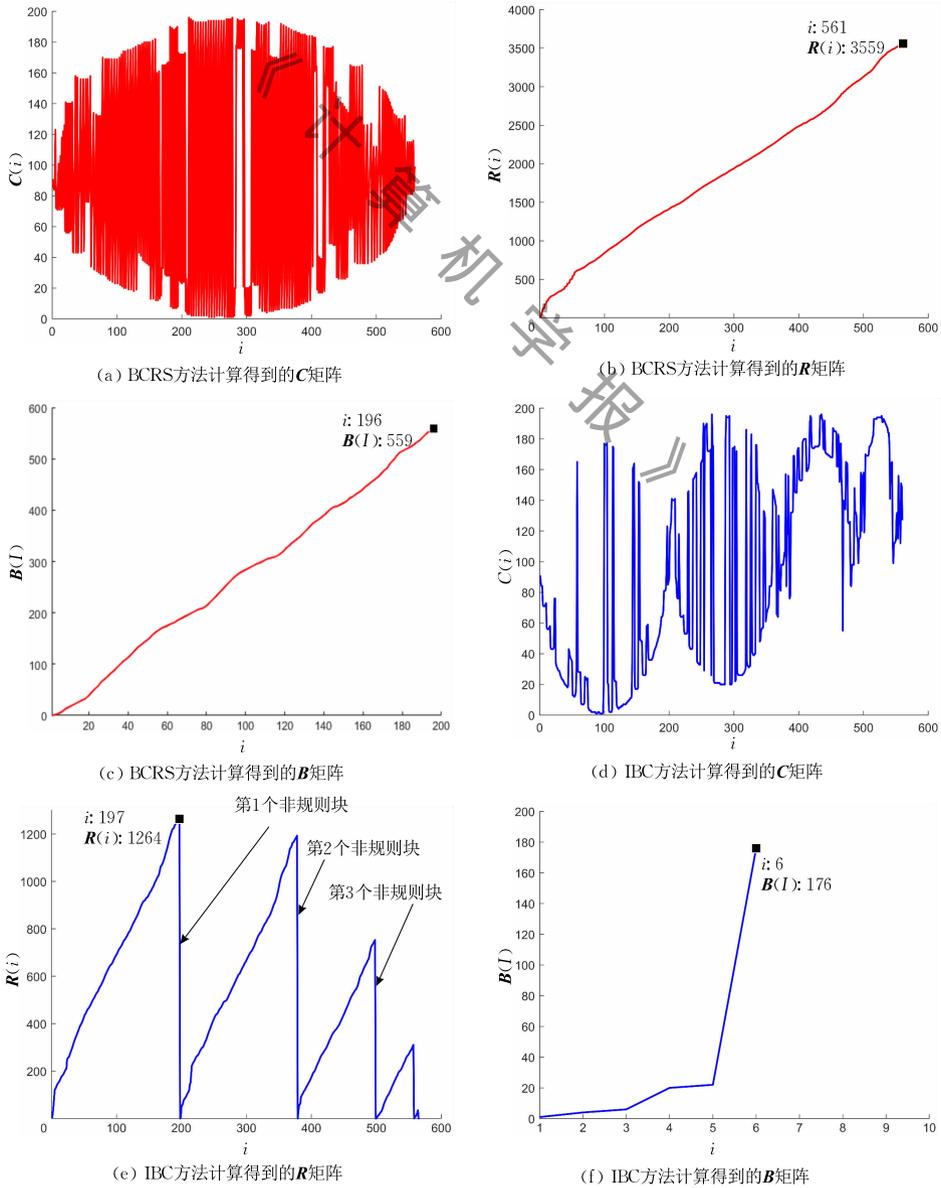


图 11 多亏格网环模型的 BCRS 与 IBC 算法的对比图

为 14.62%, 最低为 13.92%, 平均为 14.28%, 降低了算法的时间复杂度. 在占用空间改进方面, IBC 算法比 BCRS 算法节省, 最高为 22.56%, 最低为 20.72%, 平均为 21.59%, 降低了算法的空间复杂度.

图 4 所示的缩比缸体与图 8 所示的多亏格回转网环的激光成形 3D 打印实物如图 12 所示.

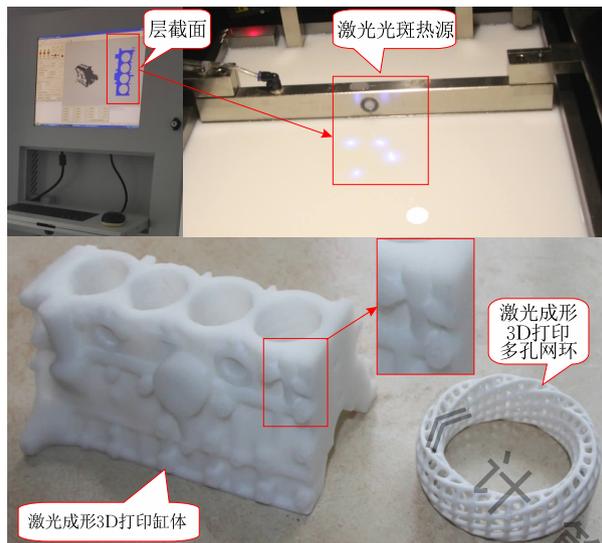


图 12 缩比缸体与多亏格网环的激光成形 3D 打印

6 结 论

(1) 提出了一种基于非规则分块压缩 (Irregular Block Compression, IBC) 的 3D 打印稀疏矩阵存储与重构方法. 根据稀疏度计算矩形规则块作为独立事件出现的概率化期望, 进一步地, 提出了稀疏矩阵非规则分块的概念. 针对稀疏矩阵的非规则连通稀疏特征, 将相邻行连通的非零块进行组合存储, 构建互连通的非规则块, 存储非零元素的数值及其有效的位置信息, 获得首行索引、首列索引、连续数目及数值集.

(2) 为了提高打印精度, 需对层截面连通域进行更高分辨率栅格化, 其后续光学转换等环节生成更多元数据, 导致非矢量化点阵数据量指数级增加, 限制了打印件尺寸. 对设定分辨率非矢量化点阵, 按照非规则块进行层截面稀疏数据重构. 通过计算相邻两层截面相似度, 对实体模型进行多层连续面打印, 减少了生成新层截面矩阵的时间, 也减少了打印机所需的存储量.

(3) 以发动机缸体和多亏格网环两种不同形态模型为例, 与传统的压缩行存储 (Compressed Row Storage, CRS) 算法和分块压缩行存储 (Block

Compressed Row Storage, BCRS) 算法相比, 在存储量改进方面, IBC 方法比 CRS 改进可达 80.60%, 比 BCRS 改进可达 14.62%, 有效降低了算法的时间复杂度; 在占用空间方面, IBC 方法比 BCRS 改进可达 22.56%, 有效降低了算法的空间复杂度. IBC 方法特别适合层截面为区块化连通的 3D 打印稀疏矩阵的复杂形态模型的 3D 打印.

未来的高效高精度 3D 打印, 还涉及三维模型层截面数据到微镜数据的映射、调制数据到激光控制器或伺服控制终端执行器的换算等核心技术. 这些都离不开海量数据压缩、数据通讯、数据存储、数据重构 (解压缩) 等计算机技术. 本文后续研究是: 在提出 IBC 方法减少层截面数据存储量和字节数上的基础上, 将 IBC 方法拓展应用到更多方面, 实现三维流形模型、打印机存储器、解码器、控制芯片等数据流的无损高效处理.

参 考 文 献

- [1] Steuben J C, Iliopoulos A P, Michopoulos J G. Implicit slicing for functionally tailored additive manufacturing. *Computer-Aided Design*, 2016, 77: 107-119
- [2] Fafenrot S, Grimmelsmann N, Wortmann M, Ehrmann A. Three-dimensional (3D) printing of polymer-metal hybrid materials by fused deposition modeling. *Materials*, 2017, 10(10): 1199-1212
- [3] Parandoush P, Lin D. A review on additive manufacturing of polymer-fiber composites. *Composite Structures*, 2017, 182(15): 36-53
- [4] Wang P, Sin W J, Nai M L S, Wei J. Effects of processing parameters on surface roughness of additive manufactured Ti-6Al-4V via electron beam melting. *Materials*, 2017, 10(10): 1121
- [5] Suryawanshi J, Prashanth K G, Ramamurty U. Tensile, fracture, and fatigue crack growth properties of a 3D printed maraging steel through selective laser melting. *Journal of Alloys and Compounds*, 2017, 725: 355-364
- [6] Zhang Xuejun, Tang Siyi, Zhao Hengyue, et al. Research status and key technologies of 3D printing. *Journal of Materials Engineering*, 2016, 44(2): 122-128
- [7] Liu Li-Gang, Xu Wen-Peng, Wang Wei-Ming, et al. Survey on geometric computing in 3D printing. *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. *计算机学报*, 2015, 38(6): 1243-1267)
- [8] Martin J H, Yahata B D, Hundley J M, et al. 3D printing of high-strength aluminium alloys. *Nature*, 2017, 549(7672): 365

- [9] Robertson I D, Yourdkhani M, Centellas P J, et al. Rapid energy-efficient manufacturing of polymers and composites via frontal polymerization. *Nature*, 2018, 557(7704): 223-227
- [10] Tumbleston J R, Shirvanyants D, Ermoshkin N, et al. Continuous liquid interface production of 3D objects. *Science*, 2015, 347(6228): 1349-1352
- [11] Matthews M J, Guss G, Drachenberg D R, et al. Diode-based additive manufacturing of metals using an optically-addressable light valve. *Optics Express*, 2017, 25(10): 11788-11800
- [12] Kelly B E, Bhattacharya I, Heidari H, et al. Volumetric additive manufacturing via tomographic reconstruction. *Science*, 2019, 363(6431): 1075-1079
- [13] Mu Q, et al. Digital light processing 3D printing of conductive complex structures. *Additive Manufacturing*, 2017, 18: 74-83
- [14] Kadry H, Wadnap S, Xu C, Ahsan F. Digital light processing (DLP) 3D-printing technology and photoreactive polymers in fabrication of modified-release tablets. *European Journal of Pharmaceutical Sciences*, 2019, 135: 60-67
- [15] He R, et al. Fabrication of complex-shaped zirconia ceramic parts via a DLP-stereolithography-based 3D printing method. *Ceramics International*, 2018, 44(3): 3412-3416
- [16] Bai Z, Demmel J, Dongarra J, Ruhe A. *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. Philadelphia: SIAM, 2000
- [17] Eberhardt R, Hoemmen M. Optimization of block sparse matrix-vector multiplication on shared-memory parallel architectures//*Proceedings of the 2016 IEEE International Parallel and Distributed Processing Symposium Workshops*. Chicago, USA, 2016: 663-672
- [18] Richter M, Runger G. Symbolic matrix multiplication for multi-threaded sparse GEMM utilizing sparse matrix formats//*Proceedings of the 2018 International Conference on High Performance Computing & Simulation*. Orleans, France, 2018: 523-530
- [19] Liu Y, Schmidt B. LightSpMV: Faster CSR-based sparse matrix-vector multiplication on CUDA-enabled GPUs//*Proceedings of the 26th IEEE International Conference on Application-Specific Systems, Architectures and Processors*. Toronto, Canada, 2015: 82-89
- [20] Choi J W, Singh A, Vuduc R W. Model-driven autotuning of sparse matrix-vector multiply on GPUs. *ACM SIGPLAN Notices*, 2010, 45(5): 115-125
- [21] Montagne E, Ekambaram A. An optimal storage format for sparse matrices. *Information Processing Letters*, 2004, 90(2): 87-92
- [22] Belgin M, Back G, Ribbens C J. Applicability of pattern-based sparse matrix representation for real applications. *Procedia Computer Science*, 2010, 1(1): 203-211
- [23] Kreutzer M, Hager G, Wellein G, et al. Sparse matrix-vector multiplication on GPGPU clusters: A new storage format and a scalable implementation//*Proceedings of the 2012 IEEE 26th International Parallel and Distributed Processing Symposium Workshops & PhD Forum*. Shanghai, China, 2012: 1696-1702
- [24] Koza Z, Matyka M, Szkoda S, Mirosław Ł. Compressed multirow storage format for sparse matrices on graphics processing units. *SIAM Journal on Scientific Computing*, 2014, 36(2): 219-239
- [25] Chalauri G, Lалуashvili V, Gelashvili K. Jagged non-zero submatrix data structure. *Transactions of A. Razmadze Mathematical Institute*, 2018, 172(1): 7-14
- [26] Xu Bin-Bin, Dai Qing-Ping, Zhu Min, Xie Duan-Qiang. Storage and computation of sparse matrix based on Huffman coding. *Computer Engineering and Science*, 2013, 35(11): 134-138(in Chinese)
(许彬彬, 戴清平, 朱敏, 谢端强. 基于哈夫曼编码的稀疏矩阵的存储与计算. *计算机工程与科学*, 2013, 35(11): 134-138)
- [27] He Xingyu, Tong Ningning, Hu Xiaowei. Dynamic ISAR imaging of maneuvering targets based on sparse matrix recovery. *Signal Processing*, 2017, 134: 123-129
- [28] Chen Xinhai, Xie Peizhen, Chi Lihua, et al. An efficient SIMD compression format for sparse matrix-vector multiplication. *Concurrency and Computation Practice and Experience*, 2018, 30(23): 4800-4809
- [29] Xu Jing-Hua, Sheng Hong-Sheng, Zhang Shu-You, et al. A method of constructing layered multi-connected domains for manifold mesh model based on adjacency topology. *Journal of Computer-Aided Design & Computer Graphics*, 2018, 30(1): 180-190(in Chinese)
(徐敬华, 盛红升, 张树有等. 基于邻接拓扑的流形网格模型层切多连通域构建方法. *计算机辅助设计与图形学学报*, 2018, 30(1): 180-190)
- [30] Xu Jing-Hua, Ren Xin-Hua, Chen Qian-Yong, et al. A slicing optimization method of manifold model for 3D printing based on dual inclinations probability. *Journal of Mechanical Engineering*, 2019, 55(13): 129-143(in Chinese)
(徐敬华, 任新华, 陈前勇等. 基于双倾斜度概率的流形模型3D打印层切优化方法. *机械工程学报*, 2019, 55(13): 129-143)



XU Jing-Hua, Ph.D., associate professor. His research interests focus on CS/CAGD.

GAO Ming-Yu, M.S. His research interest is 3D printing.

GOU Hua-Wei, M.S. His research interests is 3D printing.

ZHANG Shu-You, Ph.D., professor. His research interests focus on CAD.

TAN Jian-Rong, Ph.D., professor. His research interests include product design theory and method.

Background

The current existing 3D Printing (3DP) is commonly characterized as servo motion point by point which reduces forming efficiency. So it develops towards high efficiency and high precision, such as Digital Light Processing (DLP), Selective Laser Melting (SLM), section-by-section printing, etc. Generally, in order to improve the printing accuracy, it is necessary to rasterize the cross-section-connected domain with higher resolution whose subsequent optical conversion and other steps will generate more metadata at the same time. This in turn exponentially increases the data volume of devectorization lattice to large scale, and limit the part size directly.

The paper aims to study big data compressing for 3D printing. In this paper, a storage and reconstruction method of sparse matrix for 3D printing based on Irregular Block Compression (IBC) is proposed. Firstly, the multiple connected domain of each cross section is obtained. The layered multi-connected domain is obtained via 3D Axis-Aligned Bounding Boxes (AABB) and the Layered Cross-sectional

Mask(LCM) is generated. The rasterized lattice is constructed according to the preset resolution and converted to sparse matrix. The probabilistic expectation of rectangular Regular Blocks(RB) as independent events is calculated according to sparsity. Furthermore, the concept of Irregular Block(IB) of sparse matrix is proposed. Aiming at irregular connected sparse characteristics of sparse matrix, the connected nonzero block between adjacent rows are combinatorially stored to construct inter-connected irregular block. The value of nonzero elements and their valid location information is saved by lossless compression to obtain the first row index, the first column index, consecutive amount and value set. The sparse data reconstruction of 3D printing layered section is carried out according to irregular blocks.

This work was supported by the National Natural Science Foundation of China (Nos. 51935009,51775494,51821093), the Research and Development Projects in Zhejiang Province (Nos. 2019C01141, LGG20E050006), the National Key Research and Development Program of China (No. 2018YFB1700701).