第 31 卷 第 12 期 计算机辅助设计与图形学学报 Vol.31 No.12
2019 年 12 月 Journal of Computer-Aided Design & Computer Graphics Dec. 2019

多层次优化的网格曲面离散样条曲线设计方法

宋 滢, 金 耀*, 何利力

(浙江理工大学浙江省 2011 服装个性化定制协同中心 杭州 310018)
(fool1025@163.com)

摘 要: 为同时满足鲁棒性、高效性以及广泛适用性等要求, 提出一种基于多层次优化的网格域样条曲线设计方法. 该方法放松了曲线严格位于曲面的约束, 仅将曲线的离散控制点置于流形空间, 并采用内点法的思想, 运用基于块坐标下降法的全局优化方法进行数值求解, 最后借助局部参数化将曲线段映射到网格曲面. 在此基础上, 文中提出由粗到细的多尺度层次求解策略, 不仅能够更为准确地估算离散微分算子, 提高求解精度, 而且能够减少挪动采样点的计算量, 提升求解效率. 收敛性分析实验表明, 多层次优化方法能够快速收敛, 并在多尺度策略下获得更为光滑的结果. 和现有的投影法和光顺法相比, 该方法效率更高, 且在可控性、普适性和鲁棒性上均表现出一定的优势.

关键词: 离散样条曲线; 网格曲面; 多层次优化; 曲线设计
中图法分类号: TP391.41 **DOI:** 10.3724/SP.J.1089.2019.17559

Discrete Spline Curve Design on Surface Mesh via Cascaded Optimization

Song Ying, Jin Yao*, and He Lili

(*2011 Collaborative Innovation Center for Garment Personalized Customization of Zhejiang Province, Zhejiang Sci-Tech University, Hangzhou* 310018)

Abstract: To meet the requirements of robustness, high efficiency and wide applicability, this paper proposes a discrete spline curve design method on mesh domain based on global optimization. The method relaxes the strict manifold constraint, i.e. the curve is located on the surface by constraining discrete points of the curve. With the idea of interior point method, it adopts block coordinate descent method to solve the global optimization equation, and then maps the curve to the mesh surface through local parameterization. In addition, a multi-scales coarse-to-fine strategy is proposed to accelerate computation, which can not only estimate the discretization of the differential operator more accurately, but also decreases the computation of the sample point locations to improve the performance. Experiments on convergence analysis show that the cascaded optimization method converges quickly, obtaining smoother results with multi-scales strategy. In contrast to existing projection-based and smoothness-based methods, the proposed method has advantages in controllability, generality and robustness.

Key words: discrete spline curve; mesh surface; cascaded optimization; curve design

曲线设计是计算机图形学与计算机辅助几何设计中一项有着较长研究历史的重要课题. 目前, 针对欧氏空间的曲线设计方法已趋于成熟[1]; 随着数字几何的广泛应用以及 CAD/CAM 需求的不断增

收稿日期: 2018-10-09; 修回日期: 2019-09-16. 基金项目: 国家自然科学基金(61602416, 61702458); 浙江省公益技术应用研究计划(2017C31032); 浙江省自然科学基金(LY17F020031); 浙江理工大学科研业务费专项资金(2019Q038). 宋 滢(1981—), 女, 博士, 副教授, CCF 会员, 主要研究方向为真实感建模和绘制、几何处理; 金 耀(1984—), 男, 博士, 讲师, 硕士生导师, CCF 会员, 论文通讯作者, 主要研究方向为几何处理; 何利力(1967—), 男, 博士, 博士生导师, 主要研究方向为人机交互、大数据处理.

加, 离散网格曲面上的曲线设计逐渐受到人们关注, 并在计算几何[2]、数字几何处理[3]、虚拟手术[4]、鞋样设计[5]与数控加工[6]中得到了广泛的应用.

网格曲面上的曲线通常定义为分段线性的离散曲线. 类似于欧氏空间, 在离散意义下曲线形式亦有多种, 如测地线[2,7]、细分曲线[8]、样条曲线[9-11]等. 然而, 相比于欧氏空间的曲线, 网格曲面上的曲线由于受到流形的约束, 求解难度相对较大, 因而相应的方法不如前者成熟. 现有的方法均从不同的角度处理流形约束, 主要有如下几种.

参数化法借助参数化技术巧妙地转化流形约束, 其思想是将曲面映射到规则空间, 采用欧氏空间的传统样条曲线设计方法在参数域设计曲线, 并将其映射回原曲面. 这类方法或采用局部参数化法, 如 Lee 等[12]将主动轮廓模型引入网格域, 对局部区域进行参数化, 并在参数域上演化曲线形状; 韩林等[13]借助离散指数映射设计测地 B 样条曲线. 或采用全局参数化, 如朱文明等[14]提出基于参数化的细分曲线生成方法, 利用保角映射将不同的曲面映射到标准参数域并在其上设计细分曲线; 许晨旸等[6]和 Xu 等[15]分别借助 ABF++与 LSCM 参数化技术设计刀轨线. 然而两者均具有局限性: 局部法无法进行大范围的曲线设计; 全局法易使参数域产生较大的形变误差, 难以保证曲线的光滑性.

投影法则对流形约束进行松弛, 在欧氏空间计算曲线后迭代地将其投影到曲面上. 如基于能量最小化的样条曲线设计方法[9,16]将问题描述成带流形约束的优化问题, 采用投影梯度法进行求解, 即将松弛的空间样条曲线迭代地投影到曲面上. 该方法依赖于曲面的光滑假设, 对于网格曲面需拟合成隐函数, 否则若直接采用离散微分方法进行计算则难以保证收敛性, 且其投影步骤亦不够鲁棒. Wallner 等[8]在此基础上基于测地平均法和环绕空间法将细分曲线的线性细分法从欧氏空间推广至流形曲面, 但前者涉及测地线计算, 耗时较多; 后者涉及投影计算, 对于复杂曲面其鲁棒性降低. 之后, Morera 等[17]提出内蕴投影法, 其计算量和鲁棒性得到了较大的改善. 刘斌等[11]提出网格域的测地 B 样条插值方法, 将插值曲线采用最近点法投影到网格曲面上, 并采用反向补偿法逐步减少插值误差, 但其计算量较大, 且投影可能存在多解. 总之, 投影法易于实现, 但其鲁棒性往往较差.

光顺法则直接在流形空间求解问题, 其对光滑约束进行松弛, 在流形约束下迭代地对初始曲线进行光顺化. 这类方法的主要差异在于它们所采用的光顺策略. Jung 等[18]直接在曲面上运用主动轮廓模型演化曲线. Ji 等[3]采用新的主动轮廓模型, 结合曲线节点分裂、移动和移除操作使曲线能穿越网格面片, 提高了曲线的光滑度. Zhang 等[19]采用测地曲率流方程演化曲线, 虽然提高了计算效率, 但仍难以满足交互响应需求. Morera 等[10]基于测地细分策略推广了 de Casteljau 算法, 根据控制多边形构造测地 Bézier 曲线. Sarlabous 等[20]在此基础上, 提出了测地圆锥 Bézier 曲线设计方法, 拓宽了适用范围. Lawonn 等[4]针对网格上的曲线, 提出了一种加权拉普拉斯算子对初始曲线进行迭代式平滑. 该方法能控制曲线的光滑度和与初始曲线的贴近度, 但由于采用了局部光顺策略, 需要较多的迭代次数且不能用于构造插值曲线. 总之, 这类方法鲁棒性较好, 所设计的曲线通常具有良好的光滑性, 但是严格的流形约束使其效率降低, 同时由于所用光顺法固有的局限, 使其适用范围受到一定影响.

本文提出了一种基于层次优化的网格域离散样条曲线设计方法. 不同于传统光顺法, 本文松弛“曲线严格位于曲面”的约束, 将曲线离散成折线段, 把离散点置于流形空间并采用内点法[21]的思想, 运用块坐标下降法与投影法进行全局优化. 在此基础上, 采用由粗到细的求解策略, 提高了微分算子的离散化精度, 并能够加速算法收敛, 从而提高求解效率. 本文提出的基于多层次优化的曲线设计方法, 不仅增加了优化的自由度, 而且可便捷地施加用户控制, 使之能够设计拟合曲线和插值曲线. 此外, 相比于参数化法, 该方法能够设计大范围的曲线; 相比于投影法, 该方法鲁棒性更好.

1　离散样条曲线定义与问题描述

欧氏空间中给定 N 个型值点 $\{\boldsymbol{p}_i\}(i=1,2,\cdots,N)$, 三次样条插值曲线 $\boldsymbol{c}$ 通常定义为型值点约束下的能量函数极小值[10], 即

$$\begin{cases} \boldsymbol{c} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \int_0^1 \| \boldsymbol{x}''(t) \|^2 \, \mathrm{d}t \\ \text{s.t. } \boldsymbol{x}(t_{k_i}) = \boldsymbol{p}_i ,\ i=1,2,\cdots,N \end{cases}.$$

其中, $\boldsymbol{x}(t)$: $[0,1] \to \mathbb{R}^3$ 是曲线的参数表示形式; $\boldsymbol{x}''(t)$ 表示曲线 $\boldsymbol{x}(t)$ 关于参数 t 的二阶导数, 其与曲

率正相关；t_{k_i} 为第 i 个型值点(插值点)所对应的参数. 相应地，当 $\boldsymbol{p}_j \in S\left(i=1,2,\cdots,N\right)$ 时，光滑曲面 S 上的插值样条曲线定义为上述能量函数在流形空间 S 上的极小值，即

$$\begin{cases} \boldsymbol{c}=\arg\min\limits_{\boldsymbol{x}} \int_0^1 \|\boldsymbol{x}''(t)\|^2\,\mathrm{d}t \\ \text{s.t. } \boldsymbol{x}(t_{k_i})=\boldsymbol{p}_i,\ i=1,2,\cdots,N,\ \boldsymbol{x}(t)\in S \end{cases} \tag{1}$$

当曲面 S 为三角网格时，其上的曲线可定义为一条分段线性的离散曲线，通常可表示为一个二元组 $\boldsymbol{c}=\langle V,E\rangle$；其中，顶点集 $V=\{\boldsymbol{q}_1,\boldsymbol{q}_2,\cdots,\boldsymbol{q}_n\}\subset S$，构成曲线的边集 $E=\{\overline{\boldsymbol{q}_i\boldsymbol{q}_j}\,|\,i,j=1,2,\cdots,n\}\subset S$. 此时式(1)中所表示的曲线二阶导数需借助离散微分几何工具进行离散化，并采用数值方法进行求解.

2 本文方法

2.1 离散化

为便于式(1)中微分量的离散化，同时增加优化的自由度，本文沿用文献[10]的离散化方法，依据弦长参数化原则，对相邻 2 个插值点所形成的线段 $\boldsymbol{p}_i\boldsymbol{p}_{i+1}$ 均匀地插入 n 个点 $\{\boldsymbol{q}_j\}(j=1,2,\cdots,n)$，其中，$n=\left\lfloor \|\boldsymbol{p}_i\boldsymbol{p}_{i+1}\|/u \right\rfloor-1$，$u$ 为用户所给的平均离散步长，从而将曲线离散成一条折线段. 该策略近似地实现了曲线的均匀离散化，因此对于该曲线上的顶点 $\boldsymbol{q}_j$，其二阶微分量可离散化成一维拉普拉斯算子 $\boldsymbol{x}''(t_k)|_{\boldsymbol{x}(t_k)=\boldsymbol{q}_j}=\boldsymbol{q}_{j-1}-2\boldsymbol{q}_j+\boldsymbol{q}_{j+1}$；其中，$\boldsymbol{q}_{j-1}$ 和 $\boldsymbol{q}_{j+1}$ 分别为 $\boldsymbol{q}_j$ 的前继顶点与后继顶点的位置. 特别地，当曲线为开时，其端点处的导数可由用户自定义，本文默认设置为 0.

设该曲线为闭合曲线，且离散化后共有 M 个顶点，则式(1)可转化为

$$\begin{cases} \boldsymbol{c}=\arg\min\limits_{\boldsymbol{q}_j\in\mathbb{R}^3} F(\boldsymbol{q}_1,\boldsymbol{q}_2,\cdots,\boldsymbol{q}_M)= \\ \qquad \arg\min\limits_{\boldsymbol{q}_j\in\mathbb{R}^3} \sum\limits_{j=1}^{M} \|\boldsymbol{q}_{j-1}-2\boldsymbol{q}_j+\boldsymbol{q}_{j+1}\|^2 \\ \text{s.t. } \boldsymbol{q}_{k_i}=\boldsymbol{p}_i, i=1,2,\cdots,N, \boldsymbol{q}_j\in S, j=1,2,\cdots,M \end{cases} \tag{2}$$

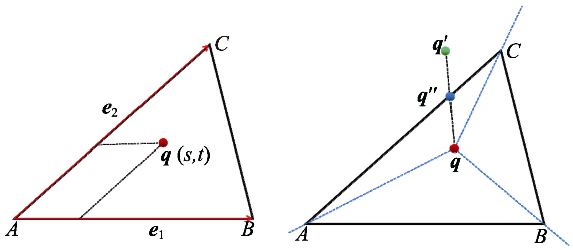
其中，$\boldsymbol{q}_M=\boldsymbol{q}_0$，$\boldsymbol{q}_{M+1}=\boldsymbol{q}_1$，下文同. 当曲线为开时，则仅需将式(2)的展开舍去第 1 项与最后 1 项即可.

2.2 数值求解

由于网格曲面 S 是分段线性的，式(2)中的流形约束 $\boldsymbol{q}_j\in S, j=1,2,\cdots,M$ 没有固定的表达式，它随着 $\boldsymbol{q}_j$ 位置的变化而变化，因而无法直接采用标准的优化方法进行求解. 为此，类似文献[8]，本文为优化目标函数引入新的变量——顶点集 $\{\boldsymbol{q}_j\}$ 所在的面片集 $\Gamma=\{T_j\}$ ($\boldsymbol{q}_j\in T_j$)，以便于描述流形约束；同时针对每个顶点 $\boldsymbol{q}_j$，在其对应的面片 T_j 上对其进行局部参数化，以消解流形约束.

设顶点 $\boldsymbol{q}_j$ 所在的面片 T_j 的 3 个顶点分别为 A，B，C，在 T_j 上建立局部坐标系 $\langle \boldsymbol{p}_0;\boldsymbol{e}_1,\boldsymbol{e}_2\rangle$：如图 1a 所示，以 $\boldsymbol{p}_0=A$ 为原点，以 $\boldsymbol{e}_1=\overline{AB}$ 和 $\boldsymbol{e}_2=\overline{AC}$ 为坐标轴. 则三角形上任意点 $\boldsymbol{q}_j$ 可参数化为

$$\boldsymbol{q}_j=\boldsymbol{q}(s_j,t_j)=\boldsymbol{p}_0+s_j\boldsymbol{e}_1+t_j\boldsymbol{e}_2,\ \ s_j,t_j\in[0,1] \tag{3}$$



a. 顶点在面片上 b. 顶点在面片外

图 1 顶点在面片上及在面片外部的局部坐标系

将曲线上的所有顶点 $\boldsymbol{q}_j$ 按式(3)进行参数化，则式(2)可重新写成

$$\begin{cases} \boldsymbol{c}=\arg\min\limits_{\{\boldsymbol{s},\boldsymbol{t}\}} F(\Gamma;\boldsymbol{s},\boldsymbol{t}) \\ \text{s.t. } \boldsymbol{q}_{k_i}=\boldsymbol{p}_i\ ,i=1,2,\cdots,N,\ s_j,t_j\geqslant 0, \\ \quad s_j+t_j\leqslant 1,\ i=1,2,\cdots,M \end{cases} \tag{4}$$

其中，$\boldsymbol{s}=[s_1,s_2,\cdots,s_M]$，$\boldsymbol{t}=[t_1,t_2,\cdots,t_M]$. 由于面片集 Γ 与参数 $\boldsymbol{s},\boldsymbol{t}$ 相互依赖，同时求解 2 组变量难度较大，为此本文采用交替迭代的方式进行求解，即固定 Γ 求解 $\boldsymbol{s}$ 和 $\boldsymbol{t}$，固定 $\boldsymbol{s}$ 和 $\boldsymbol{t}$ 更新 Γ，直至收敛.

当固定 Γ 时，式(4)中的目标函数可以进一步转化为

$$F(\Gamma;\boldsymbol{s},\boldsymbol{t})= \sum_{j=1}^{M}\|\boldsymbol{q}(s_{j-1},t_{j-1})-2\boldsymbol{q}(s_j,t_j)+\boldsymbol{q}(s_{j+1},t_{j+1})\|^2 \tag{5}$$

因此，求解 $\boldsymbol{s},\boldsymbol{t}$ 转化为计算带线性约束的优化问题.

易知，式(5)是关于 $\boldsymbol{s},\boldsymbol{t}$ 的二次函数，可转化成线性方程组进行求解. 然而，由于面片集 Γ 随着迭代而不断更新，其上曲线顶点的局部坐标系也相应更新，进而导致线性方程组的系数矩阵亦将发生变化，因此采用线性方程组求解将引起较大的时间开销. 为此，本文采用块坐标下降法[22]求解目标函数极小值，并结合投影法处理线性约束.

块坐标下降法的思想是将目标函数的变量分离成多个“块”, 然后对每个块独立求解极小值. 对于本文求解的问题, 可将每个顶点对应的参数 s_j, t_j 看做一个块. 设当前迭代次数为第 k 次, 则对于每一个非型值点 $\boldsymbol{q}_j$ (异于型值点的其他采样顶点), 式(5)可简化为

$$\begin{cases} F(\boldsymbol{q}_1^k,\cdots,\boldsymbol{q}_{j-1}^k,\boldsymbol{q}_j,\boldsymbol{q}_{j+1}^{k-1},\cdots,\boldsymbol{q}_M^{k-1}) = \| \boldsymbol{q}_{j-2}^k - \\ \quad 2\boldsymbol{q}_{j-1}^k + \boldsymbol{q}_j \|^2 + \| \boldsymbol{q}_{j-1}^k - 2\boldsymbol{q}_j + \boldsymbol{q}_{j+1}^{k-1} \|^2 + \\ \quad \| \boldsymbol{q}_j - 2\boldsymbol{q}_{j+1}^{k-1} + \boldsymbol{q}_{j+2}^{k-1} \|^2 \\ \text{s.t.}\ \ s_j, t_j \geqslant 0,\ \ s_j + t_j \leqslant 1 \end{cases} \tag{6}$$

为求解式(6)定义的优化问题, 本文采用“松弛投影法”: 首先松弛线性约束 $s_j, t_j \geqslant 0$, $s_j + t_j \leqslant 1$ 所确定的可行域, 计算目标函数的极小值, 然后将计算结果投影至可行域. 若计算所得 s_j, t_j 满足该线性约束, 则将其设置为该问题的解; 否则, 该顶点将处于面片 T_j 的外部, 将其投影至可行域边界(将顶点置于面片边界 ∂T_j), 使其依然满足线性约束. 具体讲, 设 $\boldsymbol{q}_j, \boldsymbol{q}_j'$ 分别为优化前后顶点的位置, 则将线段 $\overline{\boldsymbol{q}_j\boldsymbol{q}_j'}$ 与 ∂T_j 的交点作为投影后的新位置, 如图 1b 所示. 松弛线性约束后, 式(6)的最优解具有解析表达式, 因此可高效地求解. 其求解过程与表达式可参考附录 A.

当固定参数 $\boldsymbol{s}, \boldsymbol{t}$ 时, 可根据该参数对应顶点所在的位置更新 $\varGamma$, 即将其移至当前所在面片的邻面. 若参数 s_j, t_j 对应的顶点 $\boldsymbol{q}_j$ 位于面片内部, 即 $\boldsymbol{q}_j \in T_j \setminus \partial T_j$, 则 T_j 保持不变; 若该顶点位于面片边界, 即 $\boldsymbol{q}_j \in \partial T_j$, 此时根据该顶点的位置更新 T_j , 存在 2 种可能: (1) 当该顶点与面片的某个顶点重合时, 意味着该顶点将移入共享该顶点的某个邻面. 此时针对该顶点的所有 1-环邻接面片, 采用松弛投影法分别重新求解式(6), 把能量值较小的面片设置为当前面片, 并更新 s_j, t_j. (2) 否则, 该顶点位于面片的某条边上, 意味着该顶点将可能越过该边, 因此可将 T_j 更新为原面片相对于该边的邻面.

在迭代开始时, 首先需初始化集合 $\varGamma$. 本文根据初始路径顶点位置确定: 若顶点 $\boldsymbol{q}_j$ 位于网格边上, 则选取该边的 2 个邻接面; 若其位于网格顶点上, 则选取该顶点的 1-环邻域面, 然后分别针对选取的面片集, 分别求解式(6), 选取使能量值较小的邻接面作为对应的 T_j.

2.3　流形投影

上述算法将曲线离散成 M 个点并在流形空间进行优化, 仅保持 $V \subset S$, 而对于松弛的约束 $E \subset S$ 未在优化中考虑, 因此需将计算结果投影至流形. 本文计算曲线相邻 2 个顶点 $\boldsymbol{q}_{j-1}, \boldsymbol{q}_j$ 之间的近似测地线, 并将其作为边 $\overline{\boldsymbol{q}_{j-1}\boldsymbol{q}_j}$ 的投影. 当曲线采样足够密时, 该投影结果对曲线整体形状影响较小, 因此投影结果可认为是对优化结果的保形近似.

为提高计算效率以及鲁棒性, 本文借助局部参数化, 在参数域计算相邻 2 个顶点的欧氏路径, 并将其映射至原曲面, 以此作为近似测地线, 亦即投影结果. 具体讲, 根据相邻 2 个顶点的相对位置计算局部参数域: 若 2 个顶点位于同一面片上, 则无需投影; 否则需考虑 3 种情况:

(1) 若 2 个面片共享 1 条边, 则将该 2 面片刚性地摊平至平面;

(2) 若 2 个面片共享 1 个顶点, 则将该共享顶点的 1-环邻域面片, 按局部保角映射摊平至平面[23];

(3) 若 2 个面片无共享单形, 则运用离散指数映射法[24]计算近似测地线.

需要注意的是, 对于上述前 2 种情况, 若摊平后的局部区域为非凸, 连接 2 个顶点的欧氏路径有可能位于该区域外部(视为无效欧氏路径). 此时将内角大于 180°的“凹顶点”的 1-环邻域作为新的局部区域进行展开, 并利用情况(2)的方法重新计算该区域的参数域, 如此循环直至找到有效欧氏路径.

2.4　多层次求解策略

上述算法能够实现网格曲面上光滑离散曲线的设计, 但存在 2 个问题: (1) 曲线的离散化由插值点之间的欧氏距离近似确定, 随着曲线的演化, 其长度发生变化, 若依然采用原定的离散点将导致式(3)所示拉普拉斯算子估算不准确, 影响求解精度; (2) 从极不光滑的初始曲线演化到最终形态, 挪动点的计算频率与曲线采样密度相关, 若采样密度大, 将导致效率降低. 为此, 本文提出了由粗到精的多尺度层次求解策略——对曲线进行由粗到细采样并进行逐层优化. 该求解策略的思想在数值计算领域广泛使用, 常用于加速方程求解速度, 如多重网格法[25], 即在粗采样域上解方程, 将其作为构造细采样域上解的初值; 或用于求解非线性优化问题, 如多尺度网格变形[26], 即将对象

分解成低频与高频部分，在低频部分采用近似线性方法求解，并将解与高频合成作为最终结果. 这种策略将在求解效率和效果上均带来一定的优势. 本文方法具体描述如下:

首先进行粗化求解. 在第 2.1 节的弦长参数化阶段，对曲线进行稀疏采样. 设选取 L 个采样点 $\{\boldsymbol{f}_l\}(l=1,2,\cdots,L)$, $(L<M)$ 并按照第 2.2 节所述方法进行数值求解获得稀疏离散化的曲线 $\boldsymbol{c}'$, 并将曲线 $\boldsymbol{c}'$ 按照第 2.3 节方法投影到网格表面，该步骤通过挪动较少的采样点能够获得与最终曲线形状近似的结果.

然后进行逐层细化求解. 根据当前曲线长度，对其进行细分重采样，并重新求解式(6)的优化方程. 细化求解过程和前述类似，不同之处在于离散点更密集，其初值可以直接在曲线 $\boldsymbol{c}'$ 上进行均匀离散化，再按照第 2.2 节和第 2.3 节所述进行求解和流形投影. 由于此时所给初值更接近最终曲线，使得该策略能够快速收敛. 重复上述细化步骤，直至曲线采样密度达到用户要求.

2.5 算法步骤

本文方法具体步骤如下:

输入. 网格 S, 型值点 $\{\boldsymbol{p}_j\}_{j=1}^{N}$, 阈值 η, 最大迭代次数 T, 最大层次 L.

输出. 插值于 $\{\boldsymbol{p}_j\}$ 样条曲线 $\boldsymbol{c}\in S$.

Step1. 构造相邻型值点 $\{\boldsymbol{p}_j,\boldsymbol{p}_{j+1}\}$ 之间的 Dijkstra 路径作为初始曲线 $\boldsymbol{c}^0$, 并设置层次化级数 l=1, 弦长参数为平均边长 ρ.

Step2. repeat.

Step3. 运用弦长参数化法，由曲线 $\boldsymbol{c}^{l-1}$ 长度及参数 ρ 确定相邻型值点 $\{\boldsymbol{p}_j,\boldsymbol{p}_{j+1}\}$ 之间的当前层次离散点个数 M_l .

Step4. for each 相邻型值点对 $\{\boldsymbol{p}_j,\boldsymbol{p}_{j+1}\}$:

Step4.1. 以曲线 $\boldsymbol{c}^{l-1}$ 位置以及离散点个数 M_l 确定离散点坐标确定离散点初值;

Step4.2. 构造初始面片集 Γ.

Step5. 设置迭代次数 i =0.

Step6. repeat.

Step7. for each 离散线段:

Step7.1. 根据面片集 Γ 更新式(5)的目标函数，并求解参数 $\boldsymbol{s},\boldsymbol{t}$;

Step7.2. 根据参数 $\boldsymbol{s},\boldsymbol{t}$ 与面片集 Γ, 更新面片集 Γ.

Step8. i = i+1.

Step9. until $\|\boldsymbol{c}_i^l-\boldsymbol{c}_{i-1}^l\|\leqslant\eta$, 或 i >T.

Step10. 将 $\boldsymbol{c}^l$ 的线段投影到 S.

Step11. ρ=ρ/2, l=l+1.

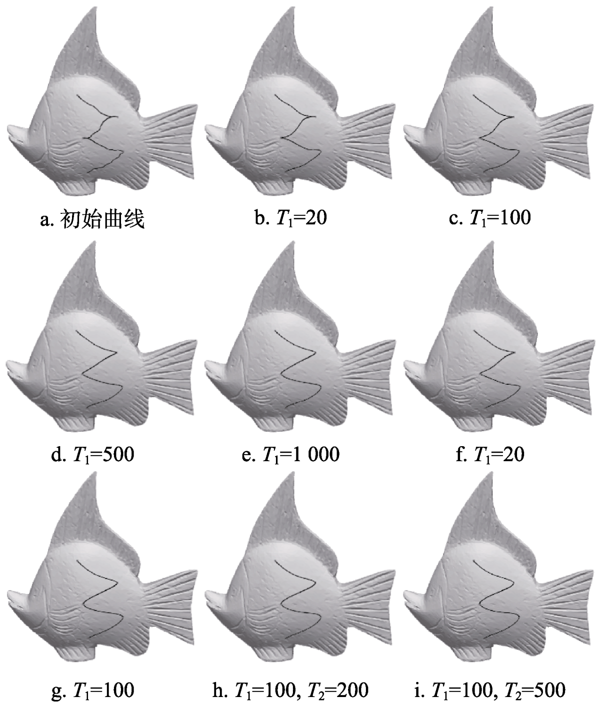
Step12. until l>L.

3 实验结果及分析

本文方法在 Intel Core i7-4760 2.1 GHz 4 核 CPU, 16 GB 内存的笔记本上基于 C++语言实现，其效率和离散化采样点的个数相关; 实验中迭代次数上限 T 为 1 000 次，曲线迭代阈值 η 设为平均边长的 10^{-6}.

3.1 收敛性分析

图 2 展示了曲线在鱼模型上不同迭代次数的结果，其中 T_i 表示第 i 层的迭代次数. 图 2a 所示为初始曲线，图 2b~图 2e 所示为常规方法(未使用层次优化)的迭代结果，图 2f~图 2g 所示为 2 级层次优化的中间结果，图 2h~图 2i 所示为 2 级层次优化的最终结果. 随着迭代次数的增加，曲线在曲面上逐渐趋于光滑并达到收敛，且层次优化方法在收敛速度上更具优势.

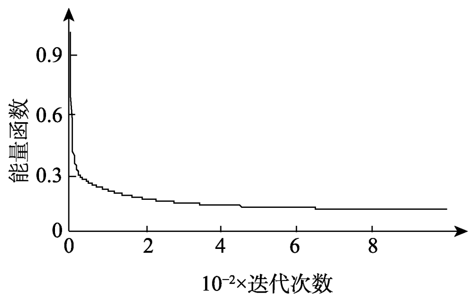


a. 初始曲线 b. T_1=20 c. T_1=100 d. T_1=500 e. T_1=1 000 f. T_1=20 g. T_1=100 h. T_1=100, T_2=200 i. T_1=100, T_2=500

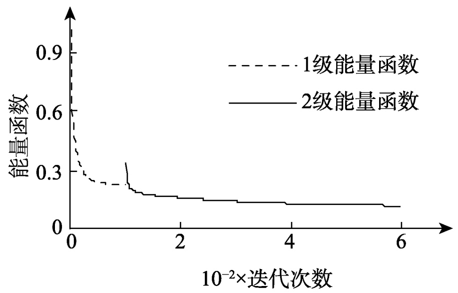
图 2 插值曲线迭代序列

图 3 给出了这 2 种方法生成的曲线能量函数值式(5)随迭代次数的变化趋势. 为体现能量变化趋势，纵坐标能量值采用了归一化处理. 随着迭代次数的增加，能量函数值总体呈快速下降趋势并收敛. 单层迭代方法在约 300 次以后趋于稳定; 而层次优化方法在第 1 级约 50 次时已经趋于稳定，可以作为初值对曲线进行第 2 级迭代求精. 虽然本文方法迭代次数较多，但由于每次迭代均采用解析式更新坐标，效率依然很高: 该曲线在未经过层次

优化的情况下(离散点个数为 105), 迭代 1 000 次仅需 23 ms, 如表 1 所示. 该实验现象验证了多尺度层次优化的特点: 当曲线采样点较疏时, 局部更新顶点坐标使用了较大的几何邻域信息, 能够大幅地挪动曲线, 从而较快地获得曲线的整体形变形状; 随着曲线采样点的加密, 更新坐标所用的几何邻域减少, 曲线挪动的幅度减小, 其效果为增加曲线的局部细节.



a. 单次迭代能量函数图



b. 多层次迭代能量函数图

图 3　鱼模型上曲线能量函数的迭代过程

表 1　模型的曲线生成时间

网格模型	顶点数/面片数	离散点数目	算法耗时/ms		
			单次/多层次	文献[4]	文献[9]
鱼	6 787/13 456	105	23/17	125	30
鸭子	9 640/19 276	139	32/22	163	97
零件	7 229/14 454	116	27/23	141	80
狮	15 000/29 996	147	38/29	171	24
兔	15 000/29 996	191	58/39	236	50
图 7a	1 753/3 514	72	14/11	84	102
图 7b	1 753/3 514	59	9/8	89	

图 4 展示了多层次优化方法比单层优化方法能够获得更为光滑的结果. 图 4a 所示为在一个平板模型上选取 4 个顶点作为曲线的初始型值点, 曲线的离散步长 u 设置为网格平均边长的 15%, 其离散采样点较为密集. 在迭代误差条件相同的情况下, 单层优化方法的曲线效果如图 4b 所示, 而多层次优化方法的曲线效果如图 4c 所示. 可见单层优化方法虽然在局部能够保证光滑, 但是在大尺度上光滑程度不如多层次优化方法. 这种多尺度优化的思想在网格形状优化方法[27]中也曾被使用, 可以避免能量极小化过程在高频分量被优化后过早停止收敛.

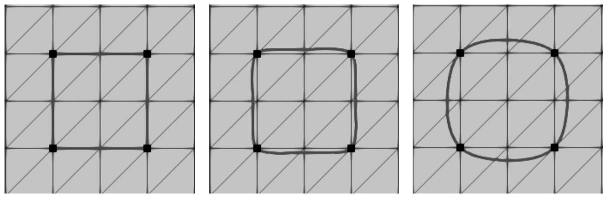
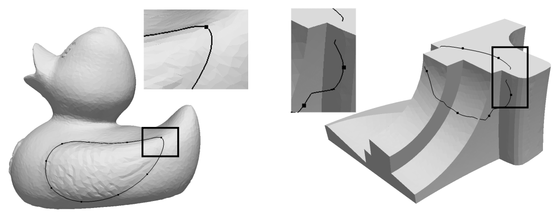

a. 初始曲线　　b. 单层方法　　c. 多层次方法

图 4　平板模型上多层次优化方法与单层方法的比较

3.2　比较实验

如图 5a 所示, 本文方法在一般模型上曲线形状能够保持光滑; 如图 5b 所示, 在具有尖锐边特征的零件模型上, 本文方法也能保证曲线的光滑性. 而文献[9]所采用的迭代法, 其迭代步长取决于局部拟合精度, 对于图 5a 所示具有尖锐特征的零件模型可能出现较大的估算误差, 从而破坏曲线的局部光滑度, 如图 5a 右侧.


a. 文献[9]

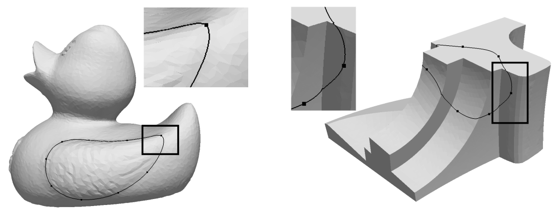
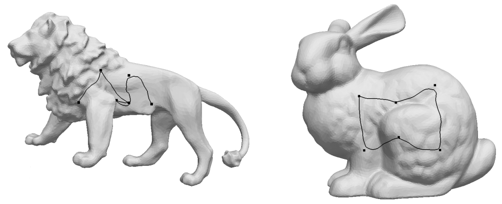

b. 本文

图 5　不同方法在鸭子和零件模型上设计曲线的对比

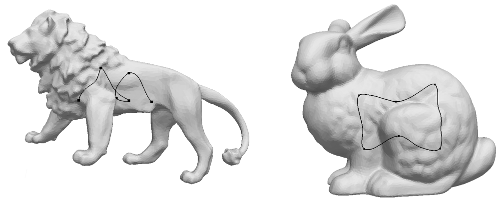
相较于基于局部光顺的曲线设计方法[4], 本文方法具有以下优势: 首先, 它可以设计插值曲线和闭合曲线, 所生成的曲线经过所有的控制点, 而文献[4]方法不能严格做到以上 2 点. 图 6a 展示了开曲线的设计结果, 可以看出文献[4]方法不能严格经过控制点, 并且无法处理曲线相交的情况; 图 5b 演示了闭曲线的设计结果, 由于文献[4]方法采用局部光顺策略, 所以在闭合曲线首尾相接处不能保证曲线光滑, 同时该方法所生成曲线的采样点无法做到自适应细分, 因此其光滑度受到网格

分辨率的影响.

本文的算法可以在多亏格模型上进行曲线设计，与模型的拓扑复杂性无关. 图 7 展示了在三环模型上设计的 2 条曲线，可环绕亏格，而基于参数化的曲线设计方法[13]由于无法对亏格区域进行参数化，因此无法设计该图所示的曲线.

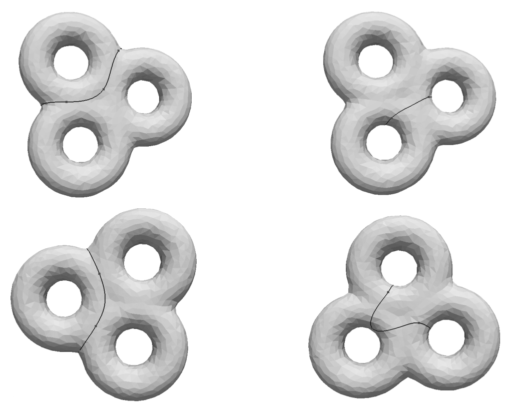


a. 文献[4]



b. 本文

图 6　不同方法对开闭曲线的处理



a. 曲线设计 1(正反面)　　b. 曲线设计 2(正反面)

图 7　亏格模型上的插值曲线

表 1 给出了几个模型的曲线生成时间以及文献[4,9]的运行时间对比. 其中离散点数目是本文方法及文献[4]采用的离散化参数，文献[4,9]的最大迭代次数为 100 次. 由于本文方法的最优化求解有解析表达(式(A4))，因此求解效率高于文献[4,9]，可满足交互应用的要求. 图 8 分析了图 6 所示兔子模型上的闭曲线在不同的离散化程度上对效率的影响，离散点越密集，计算时间越长，但平滑程度越高. 可见多层次迭代的性能优于单次迭代，且两者与离散点数量均呈线性相关趋势.

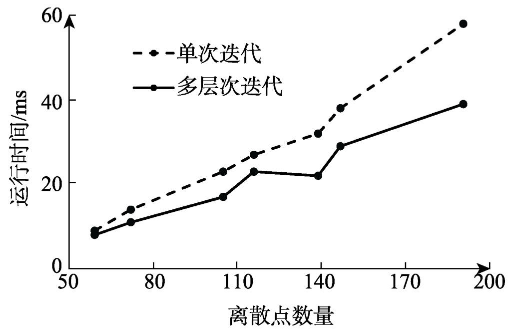


图 8　离散点数量对效率的影响

4　结　语

本文提出了一种实用高效的网格曲面曲线设计方法. 该方法松弛了曲线严格位于曲面的约束，仅对曲线离散点设置流形约束，借助内点法的思想，并采用由粗到细的多层次策略进行求解，每一层次基于块坐标梯度下降法进行全局优化. 实验表明，该方法相对于现有方法高效、鲁棒，可用于复杂拓扑网格曲面，同时能够设计闭合与开曲线、插值与光顺曲线等.

未来工作需要进一步从理论上证明本文方法的收敛性；其次，由于其复杂度与曲线离散化时采样点的个数相关，对于设计采样密度较大的曲线，其效率也会降低；因此后续工作可以尝试通过采用自适应的采样策略并优化能量方程来提高计算效率. 该方法很容易扩展到拟合曲线的设计，此时只需要将插值曲线的约束替换即可.

参考文献(References):

[1] Yan Z P, Schiller S, Wilensky G, *et al.* *K*-curves: interpolation at local maximum curvature[J]. ACM Transactions on Graphics, 2017, 36(4): Article No.129

[2] Liu Y J, Fan D, Xu C X, *et al.* Constructing intrinsic Delaunay triangulations from the dual of geodesic Voronoi diagrams[J]. ACM Transactions on Graphics, 2017, 36(2): Article No.15

[3] Ji Z P, Liu L G, Chen Z G, *et al.* Easy mesh cutting[J]. Computer Graphics Forum, 2006, 25(3): 283-291

[4] Lawonn K, Gasteiger R, Rössl C, *et al.* Adaptive and robust curve smoothing on surface meshes[J]. Computers & Graphics, 2014, 40: 22-35

[5] Liu Bin, Jiang Kaiyong. Interaction design methods of three-dimensional shoe pattern[J]. Journal of Engineering Graphics, 2010, 31(6): 117-124(in Chinese)
(刘　斌，江开勇. 三维鞋样交互设计方法研究[J]. 工程图学学报, 2010, 31(6): 117-124)

[6] Xu Chenyang, Li Jingrong, Wang Qinghui, *et al.* Tool path planning using angle-based flattening for mesh surfaces machining[J]. Journal of Computer-Aided Design & Computer

Graphics, 2017, 29(4): 728-733(in Chinese)
(许晨旸, 李静蓉, 王清辉, 等. 利用ABF++保角参数化的网格曲面刀轨规划[J]. 计算机辅助设计与图形学学报, 2017, 29(4): 728-733)

[7] Liu B Q, Chen S M, Xin S Q, *et al*. An optimization-driven approach for computing geodesic paths on triangle meshes[J]. Computer-Aided Design, 2017, 90: 105-112

[8] Wallner J, Pottmann H. Intrinsic subdivision with smooth limits for graphics and animation[J]. ACM Transactions on Graphics, 2006, 25(2): 356-374

[9] Hofer M, Pottmann H. Energy-minimizing splines in manifolds[J]. ACM Transactions on Graphics, 2004, 23(3): 284-293

[10] Morera D M, Carvalho P C, Velho L. Modeling on triangulations with geodesic curves[J]. The Visual Computer, 2008, 24(12): 1025-1037

[11] Liu Bin, Huang Changbiao, Lin Junyi, *et al.* Interpolation of geodesic B-spline curves on manifold triangulation[J]. Journal of Mechanical Engineering, 2011, 47(19): 136-142(in Chinese)
(刘　斌, 黄常标, 林俊义, 等. 流形网格曲面上测地 B 样条插值[J]. 机械工程学报, 2011, 47(19): 136-142)

[12] Lee Y, Lee S. Geometric snakes for triangular meshes[J]. Computer Graphics Forum, 2002, 21(3): 229-238

[13] Han Lin, Liu Bin. Design and reuse of curves on manifold triangulation based on discrete exponential maps[J]. China Mechanical Engineering, 2012, 23(23): 2852-2857+2863(in Chinese)
(韩　林, 刘　斌. 基于离散指数映射的网格曲面上曲线设计与复用[J]. 中国机械工程, 2012, 23(23): 2852-2857+2863)

[14] Zhu Wenming, Deng Jiansong, Chen Falai. Constructing subdivision curves on manifolds using conformal mapping[J]. Journal of Computer-Aided Design & Computer Graphics, 2007, 19(1): 48-53(in Chinese)
(朱文明, 邓建松, 陈发来. 应用保角映射构造流形上的细分曲线[J]. 计算机辅助设计与图形学学报, 2007, 19(1): 48-53)

[15] Xu J T, Sun Y W, Zhang L. A mapping-based approach to eliminating self-intersection of offset paths on mesh surfaces for CNC machining[J]. Computer-Aided Design, 2015, 62: 131-142

[16] Pottmann H, Hofer M. A variational approach to spline curves on surfaces[J]. Computer Aided Geometric Design, 2005, 22(7): 693-709

[17] Morera D M, Velho L, Carvalho P C. Subdivision curves on surfaces and applications[C] //Proceedings of the 13th Iberoamerican Congress on Pattern Recognition. Heidelberg: Springer, 2008: 405-412

[18] Jung M, Kim H. Snaking across 3D meshes[C] //Proceedings of the 12th Pacific Conference on Computer Graphics and Applications. Los Alamitos: IEEE Computer Society Press, 2004: 87-93

[19] Zhang J Y, Wu C L, Cai J F, *et al.* Mesh snapping: robust interactive mesh cutting using fast geodesic curvature flow[J]. Computer Graphics Forum, 2010, 29(2): 517-526

[20] Sarlabous J E, Mederos V H, Morera D M, *et al.* Conic-like subdivision curves on surfaces[J]. The Visual Computer, 2012, 28(10): 971-982

[21] Nocedal J, Wright S. Numerical optimization[M]. Heidelberg: Springer, 2006

[22] Fu X M, Liu Y, Guo B N. Computing locally injective mappings by advanced MIPS[J]. ACM Transactions on Graphics, 2015, 34(4): Article No.71

[23] Floater M S. Parametrization and smooth approximation of surface triangulations[J]. Computer Aided Geometric Design, 1997, 14(3): 231-250

[24] Schmidt R, Grimm C, Wyvill B. Interactive decal compositing with discrete exponential maps[J]. ACM Transactions on Graphics, 2006, 25(3): 605-613

[25] Yavneh I. Why multigrid methods are so efficient[J]. Computing in Science and Engineering, 2006, 8(6): 12-22

[26] Botsch M, Sorkine O. On linear variational surface deformation methods[J]. IEEE Transactions on Visualization and Computer Graphics, 2008, 14(1): 213-230

[27] Martin T, Joshi P, Bergou M, *et al*. Efficient non-linear optimization via multi-scale gradient filtering[J]. Computer Graphics Forum, 2013, 32(6): 89-100

附录 A　优化求解的解析推导

式(6)是一个以 $\boldsymbol{q}_j$ 为单变量的二次多项式求最小值的优化问题. 将式(3)代入式(6), 并假设

$$\begin{cases} T_1 = \left\|\boldsymbol{q}_{j-2}^k - 2\boldsymbol{q}_{j-1}^k + \boldsymbol{q}_j\right\|^2 = \left\|c_1 + s_j\boldsymbol{e}_1^j + t_j\boldsymbol{e}_2^j\right\|^2 \\ T_2 = \left\|\boldsymbol{q}_{j-1}^k - 2\boldsymbol{q}_j + \boldsymbol{q}_{j+1}^{k-1}\right\|^2 = \left\|c_2 - 2(s_j\boldsymbol{e}_1^j + t_j\boldsymbol{e}_2^j)\right\|^2 \\ T_3 = \left\|\boldsymbol{q}_j - 2\boldsymbol{q}_{j+1}^{k-1} + \boldsymbol{q}_{j+2}^{k-1}\right\|^2 = \left\|c_3 + s_j\boldsymbol{e}_1^j + t_j\boldsymbol{e}_2^j\right\|^2 \end{cases} \tag{A1}$$

其中, $c_1 = \boldsymbol{q}_{j-2}^k - 2\boldsymbol{q}_{j-1}^k + \boldsymbol{p}_0^j$; $c_2 = \boldsymbol{q}_{j-1}^k + \boldsymbol{q}_{j+1}^{k-1} - 2\boldsymbol{p}_0^j$; $c_3 = \boldsymbol{p}_0^j - 2\boldsymbol{q}_{j+1}^{k-1} + \boldsymbol{q}_{j+2}^{k-1}$.

对式(6)的优化求解必须满足

$$\begin{cases} \dfrac{\partial(T_1+T_2+T_3)}{\partial s_j} = 0 \\ \dfrac{\partial(T_1+T_2+T_3)}{\partial t_j} = 0 \end{cases} \tag{A2}$$

将式(A1)代入方程式(A2)合并同类项, 可以得到方程

$$\begin{cases} 6(\boldsymbol{e}_1\cdot\boldsymbol{e}_1)s_j + 6(\boldsymbol{e}_1\cdot\boldsymbol{e}_2)t_j = -(c_1+c_3-2c_2)\cdot\boldsymbol{e}_1 \\ 6(\boldsymbol{e}_1\cdot\boldsymbol{e}_2)s_j + 6(\boldsymbol{e}_2\cdot\boldsymbol{e}_2)t_j = -(c_1+c_3-2c_2)\cdot\boldsymbol{e}_2 \end{cases} \tag{A3}$$

假设 $a = -(c_1+c_3-2c_2)\cdot\boldsymbol{e}_1$, $b = -(c_1+c_3-2c_2)\cdot\boldsymbol{e}_2$, 由方程式(A3)可以得到 s_j 和 t_j 的解析表达

$$\begin{cases} s_j = \dfrac{a(\boldsymbol{e}_2\cdot\boldsymbol{e}_2) - b(\boldsymbol{e}_1\cdot\boldsymbol{e}_2)}{6((\boldsymbol{e}_1\cdot\boldsymbol{e}_1)(\boldsymbol{e}_2\cdot\boldsymbol{e}_2) - (\boldsymbol{e}_1\cdot\boldsymbol{e}_2)^2)} \\ t_j = \dfrac{b(\boldsymbol{e}_1\cdot\boldsymbol{e}_1) - a(\boldsymbol{e}_1\cdot\boldsymbol{e}_2)}{6((\boldsymbol{e}_1\cdot\boldsymbol{e}_1)(\boldsymbol{e}_2\cdot\boldsymbol{e}_2) - (\boldsymbol{e}_1\cdot\boldsymbol{e}_2)^2)} \end{cases} \tag{A4}$$