



基于全局相容最优角的平面网格优化

邹 杰, 徐溶延, 金 耀

(浙江理工大学信息学院, 杭州 310018)

摘 要: 为了改善平面三角网格的单元质量, 提高有限元分析的精度和收敛性, 提出了一种基于全局相容最优角的优化平面网格的方法。该方法首先直接以网格所有三角形的角度为优化变量, 以网格顶点的拓扑度来定义理想角度, 根据平面可嵌入条件构造全局相容性约束, 并在该约束空间中拟合理想角度; 然后采用高效的序列线性约束规划法数值求解最优角; 其次运用最小二乘共形映射, 将该角度及其拓扑关系嵌入平面欧氏空间; 最后为改善网格的最小角度量, 进一步采用自适应权重的相对角度误差能量进行局部优化。经过大量实验并与主流平面网格优化方法进行比较, 结果表明: 该方法与网格初始几何坐标无关, 且其优化结果的最大角度量在大多数情况下优于已有方法, 同时能够获得在平均意义上更好的三角形形状。

关键词: 平面网格; 网格优化; 全局相容; 理想角度; 平面嵌入

中图分类号: TP37

文献标志码: A

文章编号: 1673-3851 (2021) 09-0675-10

Planar mesh optimization based on globally-compatible optimal angles

ZOU Jie, XU Rongyan, JIN Yao

(School of Information Science and Technology, Zhejiang Sci-Tech University,
Hangzhou 310018, China)

Abstract: In order to optimize the element quality of planar triangle mesh and improve the accuracy and convergence of finite element analysis, this paper proposes a new planar mesh optimization method based on globally-compatible optimal angles. The proposed method first directly takes angles of all triangles of the mesh as the variants, defines the ideal angle by the topological degree of the vertices of the mesh, builds globally-compatible constraints according to the planar embedding conditions and fits the ideal angle in the constraint space. The optimal angle is then solved by using the efficient sequential linearly constrained programming. Secondly, this angle and its topological relationship are embedded into the planar Euclidean space via least square conformal mapping. Finally, in order to improve the minimal angle of the mesh, a local optimization method is further adopted based on the adaptive weighted relative angle error energy. After a lot of experiments and comparison with the mainstream planar mesh optimization methods, the results show that the proposed method is independent of the initial geometric coordinates of the mesh. In most cases, the maximum angle obtained by the proposed optimization method is superior to that of existing methods, and can achieve better triangle shapes on average.

Key words: planar mesh; mesh optimization; globally-compatible; ideal angle; planar embedding

0 引言

三角网格以三角形面片为基本单元对空间进行剖分得到,是数字几何处理与数值仿真计算中常用的离散形式。对于平面三角网格而言,其剖分技术较为成熟,通常可通过 Voronoi 图与 Delaunay 三角化等相关方法获得^[1-2]。若通过网格剖分得到的三角网格质量较差,如存在狭长三角形,甚至存在退化与翻转单元,将直接影响计算的性能与精度,使得计算结果不可靠,此时通常需要采用网格优化技术来改善网格单元质量,使其单元形状尽可能趋于正三角形。

网格优化是指对顶点位置进行几何优化,即在不改变网格拓扑连接与形状的前提下,改善网格的单元形状。网格优化的方法很多,它们大多以正多边形为参考单元来定义质量度量,并通过优化技术,使网格顶点分布趋于均匀或单元形状趋于参考单元。针对平面网格优化问题的方法大致分为拉普拉斯平滑法、基于形状度量的优化法以及基于角度的优化法 3 类,这些方法通过设计网格质量度量,利用全局或局部的数值求解技术,改善网格质量。

拉普拉斯平滑法通过局部迭代的方式将每个内部网格顶点移动至其 1-环邻域形成的多边形的中心位置^[3-4]。该方法简单高效,被广泛使用,且在后续被不断改进,出现了很多变种,如:李毅等^[5]提出了在顶点邻域复杂多边形内寻找最优位置的改进方法;Peng 等^[6]提出了一种基于三角形内接圆圆心的移动顶点策略;Yang 等^[7]提出了基于马尔可夫的平滑法,其将顶点挪动看作马尔可夫过程;Mittal 等^[8]将优化方法与拉普拉斯平滑法结合,能够获得较高质量的网格。但是,这类方法往往难以将顶点移至最佳位置,并且可能出现翻转单元。

基于形状度量的优化法通常以正多边形作为理想单元,设计偏离理想单元的形变误差度量并以此作为目标函数进行优化,如:Knupp 等^[9]采用一种矩阵条件数度量网格质量,该度量的数值变化直接反映了对网格分析方法的影响,但该方法求解较为复杂;Munson^[10]将平均比例倒数作为网格质量度量以优化平面网格,Jiao 等^[11]证明了该度量的局部凸性,并将其推广至四面体网格;Lipman^[12]从有界形变误差映射的角度,将形变误差的上下界作为约束条件并进行凸化求解,通过约束形变区间来达到优化网格的目的;Fu 等^[13]定义了可度量共形形变的高级 MIPS 度量,该方法局部为凸且具有局

部防翻转功能;Durand 等^[14]对各类网格单元定义了形状度量,并基于弹性力学理论优化网格。这类方法往往可以有效地控制网格的质量或形变度量,但是当网格拓扑连接不规则程度较高时,其优化效果有限。

基于角度的优化法则从优化网格的角度分布的思路进行优化,如:Zhou 等^[15]从平分顶点 1-环邻域多边形内角出发,分别计算该顶点在每个角平分线上的最佳位置,将所有位置进行加权平均得到顶点位置;Xu 等^[16]在此基础上,利用平分角定义等价的距离误差度量,并采用 Levenberg-Marquardt 方法求解,获得了更好的优化结果;李宗领等^[17]提出了一种最小角最大化方法对网格进行几何优化,验证了角度作为网格质量度量的重要性;Renka^[18-19]以最小化网格单元面积平方和为优化目标,当权重设置为顶点到邻域对边距离的倒数时,能够获得很好的角度优化效果,算法鲁棒性强,优于主流方法。但是这类方法均直接以顶点坐标作为变量,往往对初值较为敏感,若初值较差,如出现大量单元翻转,则会使得迭代收敛变慢,甚至难以得到理想的结果。

网格中的最小角和最大角能够有效且直接地反映全局质量和单元形状,本文提出一种基于角度的平面网格优化方法优化网格质量,实现网格最小角最大化和最大角最小化。与其他同类方法^[15-19]不同,本文首先直接以角度作为变量,利用网格的拓扑连接信息定义理想角度,并以二次拟合误差作为目标函数;然后在有效的角度空间构造全局相容性约束进行非线性优化;最后从求解出的最优角度信息重建出网格的几何坐标。该方法由于仅利用了网格的拓扑信息,因此求解时与网格的几何坐标初值无关,能够在全局相容性约束下获得角度分布最优的网格。

1 方法

针对在角度空间中的网格优化问题,本文首先利用网格拓扑构造了理想的角度拟合目标,并设计了角度空间中的网格表达;然后将该表达转化为角度相容性条件组成带约束的非线性优化问题;再结合序列线性约束规划法和牛顿法数值求解最优角,最后利用最小二乘重建网格几何,并结合局部优化策进一步提高网格质量。

1.1 理想角度构造

平面三角网格 M 可以表示为 $\{V, T\}$, 其中: $V = \{v_1, \dots, v_n\} \in \mathbf{R}^2$ 表示 n 个顶点构成的网格几何;

$T = \{t_1, \dots, t_m\}$ 为 m 个三角形构成的网格拓扑, 每个三角形 t_i 由 V 中三个顶点组成。每个三角形 $t \in T$ 的每个顶点 $v \in t$ 的内角 α_v^t 组成的集合为 $\{\alpha_v^t\}$, 其中: 角度集 $\{\alpha_v^t\}$ 元素个数为 $3m$ 。本文直接以角度为优化变量, 通过构造网格的理想角度分布 $\{\beta_v^t\}$ 来设计网格的全局角度优化度量, 以逼近目标:

$$\min E(A) = \min \sum_{t \in T} \sum_{v \in t} w_{t,v} (\alpha_v^t - \beta_v^t)^2 \quad (1)$$

其中: A 为角度集 $\{\alpha_v^t\}$ 组成的大小为 $3m$ 的角度向量; β_v^t 为对应第 t 个三角形中关于顶点 $v \in t$ 的内角 α_v^t 的最优角; $w_{t,v}$ 为权重, 默认情况下设为均权值, 即对于所有的 $w_{t,v} = 1$ 。

在现有的网格优化方法中, 通常以正三角形作为理想单元并定义单元质量度量^[9-14], 因此对应的理想角度分布为:

$$\beta_v^t = \frac{\pi}{3} \quad (2)$$

事实上三角网格的角度分布与网格的拓扑连接直接相关, 当网格顶点非正则时, 即顶点邻接点的个数不为 6 时, 在理想状态下其邻接角度无法均达到 $\pi/3$ 。因此考虑网格拓扑对角度分布的影响, 本文提出了一种拓扑度相关的理想角:

$$\beta_v^t = \frac{1}{|N(v)|} \gamma_v \quad (3)$$

其中: $|N(v)|$ 表示顶点 v 的拓扑度; γ_v 为顶点 v 的周角之和。这种根据网格顶点的度数确定理想角的思想已在六面体网格优化中得到了应用, 并获得了良好的效果^[20]。

但是, 以式(3)作为理想角度并未考虑角度之间的全局相容性, 由此形成的角度分布往往不能较好地满足三角形内角和约束。因此, 本文将三角形内角和为 π 这一约束加入理想角相容性条件, 对于每个三角形 t , 设由式(3)确定的三个顶点 β_v^t 的理想角之和为 $\beta_t = \sum_{v \in t} \beta_v^t$, 则该三角形 t 中任意顶点的内角对应的理想角为:

$$\bar{\beta}_v^t = \frac{\beta_v^t}{\beta_t} \pi \quad (4)$$

即式(4)对式(3)进行修正, 使其满足内角和约束。

1.2 全局相容性约束描述

二维网格的角度集确定一个三角剖分。但式(1)所确定的角度空间 $\{\alpha_v^t\}$ 未考虑网格的拓扑与几何约束, 一般情况下无法嵌入平面欧氏空间。因此, 本文借鉴基于角度的参数化 (Angle-based flattening, ABF) 方法^[21-22], 针对网格优化问题构造网格角度的

全局相容性约束, 并结合目标函数式(1), 将其作为硬约束进行求解。与 ABF 方法不同, 本文考虑的网格优化问题需固定边界顶点坐标, 需额外构造针对边界顶点的约束。因此, 在角度空间中求解需满足如下全局相容性约束:

a) 三角形内角的相容性约束。设对于网格中任意三角形 t 关于顶点 $v \in t$ 的内角 α_v^t , 则由三角形内角和为 π , 得到约束条件:

$$\forall t \in T, C_{\text{Tri}}(t) = \sum_{v \in t} \alpha_v^t - \pi = 0 \quad (5)$$

b) 顶点周角的相容性约束。对于任意网格顶点, 其 1-环邻域内角和是一个常数。网格内部和边界点局部示意图如图 1 所示。对所有的内部顶点集合 $V_{\text{int}} (\forall v \in V_{\text{int}})$, 顶点 v 的 1-环邻域的内角和为 2π ; 对于所有边界顶点集合 $V_{\text{bnd}} (\forall v \in V_{\text{bnd}})$, 其顶点 v 的 1-环邻域内角和保持不变, 即 $\gamma_v = \sum_{t \in N(v)} \alpha_v^t$ 。因此, 该约束可统一表示为:

$$\forall v \in V, C_{\text{plan}}(v) = \sum_{t \in N(v)} \alpha_v^t - \gamma_v = 0 \quad (6)$$

c) 网格重建的相容性约束。在三角网格中, 一条边可被两个三角形共享, 因此需确保角度空间中的任意两个相邻三角形共享边的长度相等。如图 1(a)所示, 在满足前述两个约束条件下, 若不添加该约束, 则可能出现重建后两个三角形共享边的边长不等的情况。因此, 本文构造如下有效重建的约束条件:

$$\forall v \in V, C_{\text{Re}}(v) = l_{\text{left}} * \prod_{t \in N(v), v_{\text{next}} \in t} \sin \alpha_{v_{\text{next}}}^t - l_{\text{right}} * \prod_{t \in N(v), v_{\text{prev}} \in t} \sin \alpha_{v_{\text{prev}}}^t = 0 \quad (7)$$

网格内部和边界顶点局部示意图如图 1 所示, 对网格中任意一个三角形 t 的任意一个顶点 $v \in t$, 都能在这个三角形上找到其逆时针的后一个顶点 v_{next} 和顺时针的前一个顶点 v_{prev} 以及对应的角度 $\alpha_{v_{\text{next}}}^t$ 和 $\alpha_{v_{\text{prev}}}^t$; 网格中任意顶点 v 以其任意一条邻接边为起点, 其顺时针的边界边为 l_{left} , 反之为 l_{right} , 当 $v \in V_{\text{int}}$ 时, $l_{\text{left}} = l_{\text{right}}$, 此时, l_{left} 与 l_{right} 可约除; 当 $v \in V_{\text{bnd}}$ 时, l_{left} 与 l_{right} 不等, 但为常数。

结合式(1)、式(5)一式(7), 拟合理想角度分布的优化问题可归结为下式:

$$\left\{ \begin{array}{l} \min E(A), \\ \text{s.t.} \quad \sum_{t \in T} C_{\text{Tri}} = 0, \\ \sum_{v \in V} C_{\text{plan}} = 0, \\ \sum_{v \in V} C_{\text{Re}} = 0 \end{array} \right. \quad (8)$$

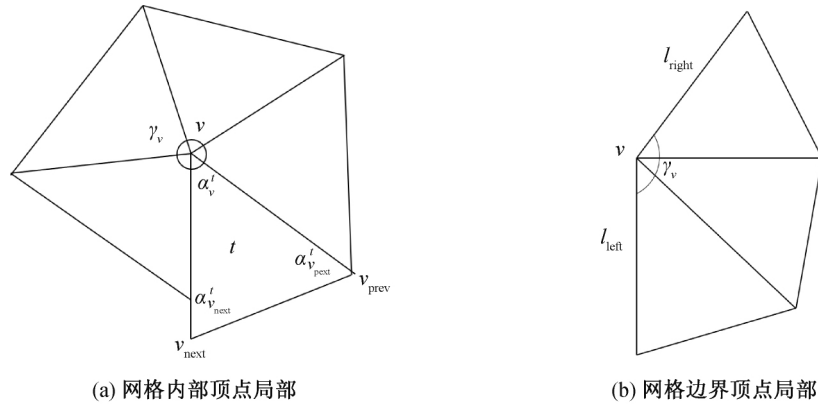


图1 网格内部和边界顶点局部示意

1.3 全局相容最优角求解

通过构造上述全局相容性角度约束,最终将角度优化问题转化为式(8)极小化问题。该式是一个带非线性约束的优化问题,因此可运用拉格朗日乘子法将其转换为无约束问题:

$$F(\mathbf{x}) = F(\mathbf{A}, \mathbf{\Lambda}_{\text{Tri}}, \mathbf{\Lambda}_{\text{Plan}}, \mathbf{\Lambda}_{\text{Re}}) = E(\mathbf{A}) + \sum_{t \in T} \lambda_{\text{Tri}}^t C_{\text{Tri}}(t) + \sum_{v \in V} \lambda_{\text{Plan}}^v C_{\text{Plan}}(v) + \sum_{v \in V} \lambda_{\text{Re}}^v C_{\text{Re}}(v) \quad (9)$$

其中: $\mathbf{\Lambda}_{\text{Tri}}, \mathbf{\Lambda}_{\text{Plan}}, \mathbf{\Lambda}_{\text{Re}}$ 分别对应以式(5)~式(7)为约束的拉格朗日乘子组成的向量; \mathbf{x} 是变量 \mathbf{A} 和所有拉格朗日乘子组成的向量。对于该非线性问题式(9),本文运用牛顿法进行求解,其计算流程为:

- 计算函数 F 的在当前迭代值 \mathbf{x}^k 的梯度 $\nabla F(\mathbf{x}^k)$ 与 Hessian 矩阵 $\nabla^2 F(\mathbf{x}^k)$;
- 通过方程组 $\nabla^2 F(\mathbf{x}^k) \delta = -\nabla F(\mathbf{x}^k)$ 求解下降方向 δ ;
- 通过线性搜索等步长搜索策略获取下降步长 τ ;
- 计算下一迭代值 $\mathbf{x}^{k+1} = \mathbf{x}^k + \tau \delta$;
- 给定精度 ϵ_g 与 ϵ_f , 判断当前迭代点是否位于驻点: 即当满足梯度收敛条件 $\|\nabla F(\mathbf{x}^k)\| < \epsilon_g$ 或函数值收敛条件 $|F(\mathbf{x}^k) - F(\mathbf{x}^{k+1})| < \epsilon_f$ 时, 迭代终止, 否则退回步骤 a)。

根据牛顿法的算法流程可知,在求解下降方向时先要得到 Hessian 矩阵的逆,由于式(6)的高度非线性会使得矩阵逆的计算代价非常高。为降低计算复杂度,本文采用序列线性约束规划法并采用类似 ABF++ 中的矩阵分解技术^[21]用以简化 Hessian 矩阵计算。给定顶点数 n 和三角面数 m 的网格。原线性系统式(8)的规模为 $4m+2n$ (角度变量 $3m$, 三个约束 $m+2n$), 根据欧拉公式,面片个数约等于顶点个数两倍,即 $m \approx 2n$, 通过分解技术可以减少

到 $2n+2n$, 因此系统规模大约可减少 60%。具体计算过程可参考文献[21]。

1.4 网格重建

给定网格角度集 $\{\alpha_v^t\}$, 可运用最小二乘共形映射方法^[23] 将其嵌入平面欧氏空间。设任意三角形 $t = (v_i, v_j, v_k)$, 由于三角形的 2 条边 $v_{ij} = v_j - v_i$ 与 $v_{ik} = v_k - v_i$ 在旋转与伸缩变换下重合, 结合正弦定理 $\|v_{ik}\| \sin \alpha_{v_k}^t = \|v_{ij}\| \sin \alpha_{v_j}^t$, 获得 2 条边的长度关系, 因此可得方程:

$$v_{ik} = \frac{\sin \alpha_{v_j}^t}{\sin \alpha_{v_k}^t} \begin{bmatrix} \cos \alpha_{v_i}^t & -\sin \alpha_{v_i}^t \\ \sin \alpha_{v_i}^t & \cos \alpha_{v_i}^t \end{bmatrix} v_{ij} = \mathbf{M}^t v_{ij}.$$

其中: \mathbf{M}^t 表示关于三角形 t 角度的旋转缩放矩阵。据此, 本文在最小二乘形式下构建整个网格的重建误差:

$$\min \sum_t \|\mathbf{M}^t(v_j - v_i) + v_i - v_k\|^2.$$

上述优化目标是二次函数, 因此可转化为线性方程组进行求解, 同时为避免方程退化且使解唯一, 需至少固定两个顶点。本文将网格的所有边界点固定, 以防止边界变形。

1.5 针对小角度的局部网格优化

基于前述方法能够获得趋于理想角度分布的平面网格。但是, 由于网格优化度量式(1)采用了最小二乘形式, 所得到的结果不可避免地出现一些小角度的三角形。为了改善网格优化结果, 本文依次搜索所有小于给定阈值的角度 $\bar{\gamma}$, 并以该角所在三角形的顶点为中心搜索其 N -环邻域, 然后基于前述算法对该邻域所形成的局部三角形网格进行优化。为了有效地惩罚小角度, 本文从两方面对网格优化度量式(1)进行修正:

- 提高理想角度 β_v^t 。小角度往往出现在拓扑度较高的顶点邻域中, 因此适当加大理想角度有助

于该角度的提高。因此,本文设置理想角度为: $\beta_v^t = \max\left\{\beta_v^t, \frac{\pi}{3}\right\}$, 并采用式(4)进行校正。

b) 设置自适应权重 $w_{t,v}$ 。设当前网格的角度为 $\{\alpha_v^t\}$, 为使小角度尽可能逼近其理想角, 可增大其对应能量项的权重, 如将权重设置为当前网格角度的反函数: $w_{t,v} = \frac{\beta_v^t}{\alpha_v^t}$; 同时为减少大理想角度和小理想角逼近误差的差异, 将式(1)的绝对差修改为相对差形式, 即: $\left(\frac{\alpha_v^t - \beta_v^t}{\beta_v^t}\right)^2$ 。综上所述, 局部网格优化的目标函数修正为:

$$\min E(A) = \min \sum_{t \in T} \sum_{v \in t} \frac{1}{\alpha_v^t \beta_v^t} (\alpha_v^t - \beta_v^t)^2.$$

2 实验及结果分析

本文采用 C++ 语言实现了上述平面网格优化算法, 其中所涉及的矩阵计算使用 Eigen 库, 求解大型稀疏线性方程组采用了其中的乔利斯基分解法接口, Eigen: SimplicialLDLT, 角度优化的初值设置为理想角 $\{\beta_v^t\}$, 小角度阈值 $\bar{\gamma} = 5^\circ$, 局部优化选取的邻域 $N = 4$ 。

为验证算法的有效性, 本文进行了一系列对比实验, 并对结果运用文献[19]所采用的最小角、最大角、等度量进行量化评估。其中本文对比了 5 种具有代表性的平面网格优化方法: 拉普拉斯平滑^[3], 记作 A; 基于距离的角度度量法^[16], 记作 B; 有界形变映射法^[12], 记作 C; 基于加权面积平方和的角度度量法^[19], 记作 D; 基于物理的形状匹配度量法^[14], 记

作 E。其中算法 A、B、E 采用 C++ 语言实现, 算法 C、D 采用作者提供的 Matlab 实现。

为验证不同理想角策略(式(2)一式(4))对结果的影响, 本文测试了 10 个以正方形为边界的网格, 分别比较三种理想角下的优化结果, 结果见表 1。其中正方形的每条边均等采样 20 个点, 在内部随机采样 500 个点, 共计 580 个顶点, 并使用带约束的德劳内三角化 (Constrained Delaunay triangulation, CDT)^[24] 生成三角形网格, 且各个网格的拓扑连接关系均互不相同。表 1 内各个单元括号内三个数据表示单元质量的度量, 分别为内切圆与外接圆的半径最小比值 $q_{\min} = 2r_{\text{in}}/r_{\text{out}}$ (该值越接近 1 表示越逼近等边三角形, 越接近 0 表示越退化), 网格的最小角 α_{\min} 与最大角 α_{\max} , 由此构成了三元组 $(q_{\min}, \alpha_{\min}, \alpha_{\max})$ 。图 2 为不同理想角策略下的 0 号正方形网格优化的结果。由表 1 和图 2 可见, 理想角度按照式(4), 即考虑了角度相容性时, 网格质量优化效果最佳。

表 1 不同最优角策略下的正方形质量优化结果

编号	等边三角形 $\beta_v^t = \frac{\pi}{3}$	顶点度平均 $\beta_v^t = \frac{1}{ N(v) } \gamma_v$	等比例分配 $\beta_v^t = \frac{\beta_v^t}{\beta_t} \pi$
0	(0.00 0.01 159.27)	(0.00 0.18 175.48)	(0.30 12.07 129.01)
1	(0.32 10.24 126.58)	(0.32 10.24 126.58)	(0.36 13.28 125.11)
2	(0.23 8.38 129.80)	(0.00 0.32 162.40)	(0.27 10.26 131.93)
3	(0.31 12.11 126.48)	(0.31 12.11 126.48)	(0.32 13.61 129.65)
4	(0.00 0.22 177.39)	(0.00 0.01 178.76)	(0.27 10.93 132.51)
5	(0.27 9.97 129.39)	(0.27 9.97 129.39)	(0.28 14.11 132.28)
6	(0.28 11.34 128.33)	(0.28 11.34 128.33)	(0.27 14.17 126.12)
7	(0.35 14.13 127.67)	(0.35 14.13 127.67)	(0.31 15.14 130.74)
8	(0.21 7.62 130.79)	(0.21 7.62 130.79)	(0.28 10.28 133.54)
9	(0.21 7.45 130.11)	(0.21 7.45 130.11)	(0.31 11.36 132.04)

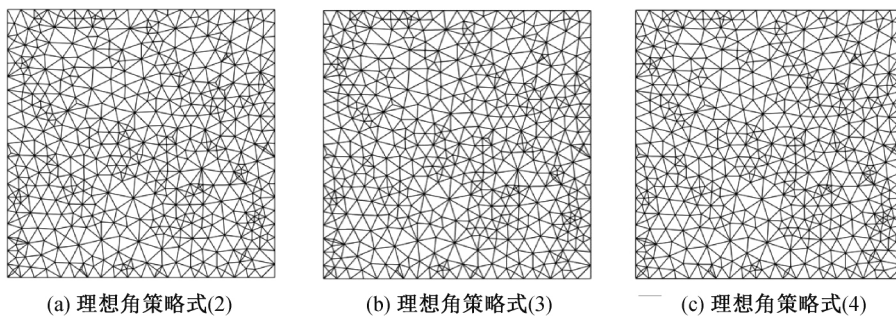


图 2 不同理想角策略下的 0 号正方形网格优化结果

本文算法在两种收敛条件(梯度收敛条件和函数值收敛条件)下, 均能达到收敛状态, 且梯度收敛的情形更多。为观察算法的收敛行为, 本文分别绘制了两种不同收敛条件下的能量函数曲线, 如图 3 所示。对于 Apple 模型, 由于给定初值拟合理想角

度效果好但全局相容条件吻合度较差, 其能量函数逐渐上升使其逐步满足相容性条件, 但随后便呈现下降趋势并趋于稳定, 其梯度模长变化亦呈现类似的变化趋势; 而对于 Bone 模型, 能量函数呈现单调下降趋势, 其梯度模长变化总体上也趋于下降, 并最

终将为零。此外,在两种收敛条件下,算法在绝大多数情况下均能较好地满足全局相容的等式约束,因而所获得的角度分布能够几乎零误差地重建网格。

即使在极个别情况下,全局相容的等式约束无法得到满足,运用 1.5 节最小二乘重建方法也能获得较为理想的结果。

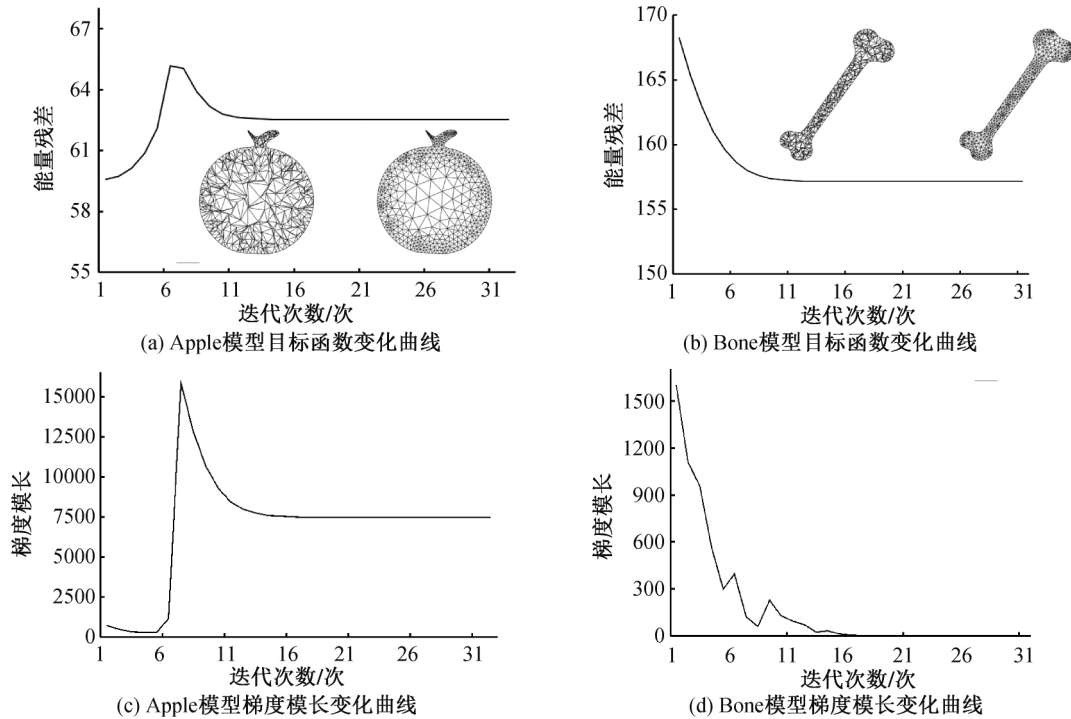


图3 苹果模型与骨头模型的目标函数与梯度模长变化曲线

本文算法在多数情况下仅需全局优化便可获得较好的结果,而优化结果一旦出现小角度,通过执行局部优化能够有效地改善最小角度网格质量。本文挑选了两个经过全局优化而出现小角度的例子,通过局部优化后,其结果见图4—图5。其中Puzzle网格模型在局部优化前的质量为(0.00 0.14

179.54),而优化后变为(0.01 7.83 149.11); Telescope网格模型在局部优化前的质量为(0.00 0.01 179.96),而优化后变为(0.04 6.89 162.81),可见其最小角均有明显提升。同时,由局部放大图可见,经过局部优化,优化前的狭长三角形形状得到了改善。

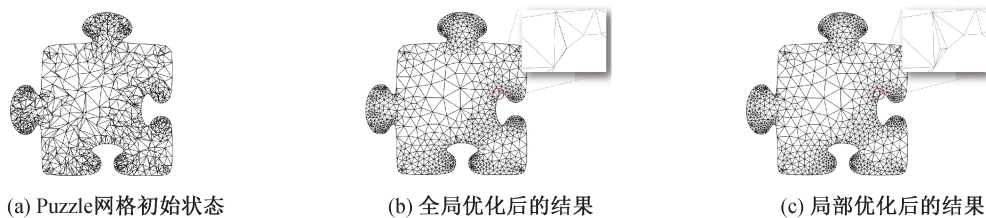


图4 Puzzle网格模型经过局部优化前后的示例图像



图5 Telescope网格模型经过局部优化前后的示例图像

本文提出的算法,由于仅与网格拓扑连接相关,与网格的初始几何坐标无关,因而即使当输入网格

杂乱无章、出现大量自交时,本文算法依然有效且不影响效率。传统基于顶点坐标的优化方法对初值较

为敏感,特别是当优化目标函数是非线性非凸时,若所给的初值较差(如网格单元大量自交)将使迭代次数增加,收敛变慢,甚至难以获得理想的优化结果。本文分别与算法 B 和算法 D 进行对比,如图 6—图 7 所示,算法 B 由于采用了非线性优化,当初值较差时(图 6(a)、图 7(a)),陷入了局部极小值解,优

化结果较差(图 6(c)、图 7(c));算法 D 虽然对于较差(图 6(a)—(b))与较好(图 7(a)—(b))的初值,均能获得理想的优化结果,但不同的初值影响了算法的迭代次数。对于图 6 的例子,其迭代次数分别为 15 和 24;而对于图 7 中的例子迭代次数分别为 14 和 28。

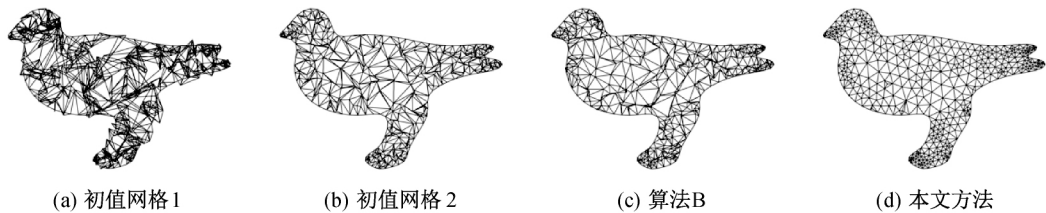


图 6 不同初值下的网格 bird 以及方法 B 和本文方法的优化图像

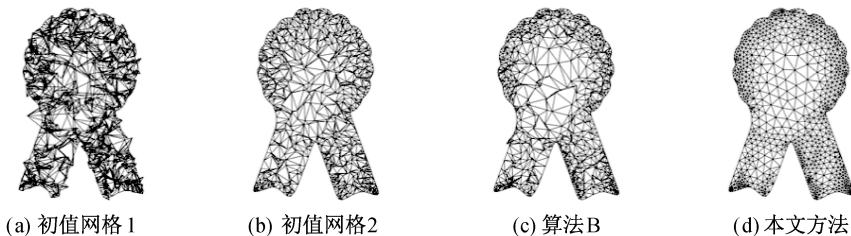


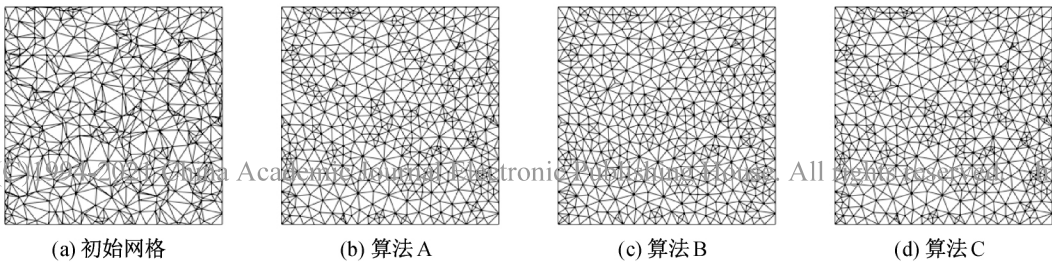
图 7 不同初值下的网格 medal 以及方法 B 和本文方法的优化图像

为了充分展示算法的优点,本文比较了 A、B、C、D、E 这五种网格优化算法,优化后网格的质量指标见表 2,本文算法的相关数据见表 1 最后一列。由此可见,本文算法的最大角具有一定优势,除了个别比算法 E 大,总体均小于其他算法的结果;其内

切圆与外接圆半径的平均比值较高,均大于 0.27,且在大多数情形下亦比其他方法更优;其最小角度量虽然优势不太明显,但均大于 10°,总体上比拉普拉斯算法与有界形变优化算法好。图 8 为第 0 号网格的优化结果对比图。

表 2 各个算法在正方形网格上测试所得到的优化结果

编号	初始网格	算法 A	算法 B	算法 C	算法 D	算法 E
0	(0.00 0.18 179.36)	(0.13 13.90 149.68)	(0.27 18.75 135.54)	(0.13 13.97 149.55)	(0.28 19.96 134.82)	(0.33 17.94 130.73)
1	(0.01 0.50 172.80)	(0.20 12.26 143.07)	(0.34 17.70 127.23)	(0.20 12.26 142.94)	(0.37 20.70 125.58)	(0.36 17.31 126.38)
2	(0.01 0.92 172.08)	(0.13 13.52 150.39)	(0.25 18.57 138.04)	(0.13 13.52 150.37)	(0.29 20.90 134.37)	(0.30 18.26 132.40)
3	(0.00 1.08 177.01)	(0.15 15.78 147.77)	(0.28 17.11 135.38)	(0.15 15.72 147.63)	(0.28 20.41 134.94)	(0.35 18.15 133.21)
4	(0.00 0.70 178.10)	(0.06 7.55 159.44)	(0.21 13.63 141.26)	(0.06 7.47 158.91)	(0.22 16.94 140.57)	(0.27 12.96 137.99)
5	(0.00 0.05 179.68)	(0.08 11.46 156.42)	(0.30 21.36 133.38)	(0.08 11.47 156.39)	(0.24 20.65 138.41)	(0.33 18.78 138.18)
6	(0.00 0.36 179.14)	(0.10 11.40 154.14)	(0.35 18.19 129.30)	(0.10 11.51 154.02)	(0.35 20.41 128.70)	(0.35 16.83 133.61)
7	(0.00 0.16 176.08)	(0.13 14.58 150.40)	(0.32 20.77 131.32)	(0.13 14.60 150.37)	(0.28 19.33 135.05)	(0.35 19.06 135.46)
8	(0.00 0.10 179.69)	(0.11 12.66 152.64)	(0.22 17.30 140.50)	(0.11 12.92 152.94)	(0.26 19.35 136.96)	(0.32 18.33 133.57)
9	(0.01 1.10 170.42)	(0.07 9.17 157.20)	(0.19 11.08 138.64)	(0.08 9.27 156.99)	(0.30 19.13 132.91)	(0.25 14.04 136.01)



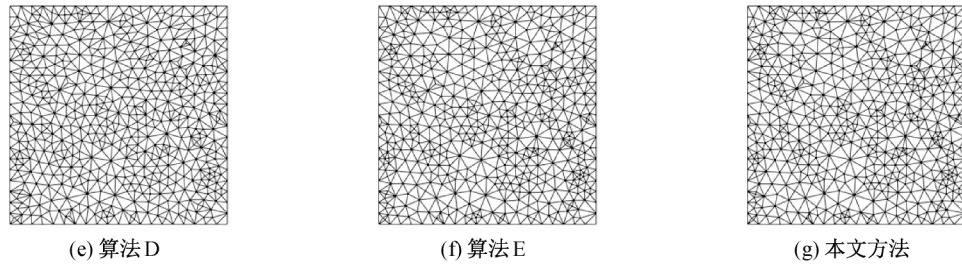


图 8 各个算法对正方形网格的优化图像

本文测试了一般形状的二维网格,并对其优化结果进行了对比。利用一些二值图像轮廓生成多边形边界,然后同样运用 CDT 在其轮廓内生成网格,并运用上述算法对其进行优化,结果见表 3 与图 9。

可见,实验结果总体上与正方形网格的测试例子类似,但是最大角度量优势明显,均比其他方法更优。说明本文提出的方法能够最大程度上减少最大角,同时能够较好地改善网格单元的形状。

表 3 各类轮廓网格优化结果

网格名称	初始网格	算法 A	算法 B
Crown	(0.00 0.06 179.67)	(0.05 6.64 162.25)	(0.22 7.48 140.03)
Bell	(0.00 0.08 179.22)	(0.06 9.31 159.05)	(0.10 10.46 152.72)
Face	(0.00 0.02 179.87)	(0.18 16.70 144.49)	(0.15 11.08 144.64)
Flatfish	(0.00 0.08 179.76)	(0.12 10.90 150.38)	(0.19 11.73 142.45)
Gear	(0.00 0.01 179.85)	(0.17 15.26 145.04)	(0.27 18.45 134.99)
Ping-pong	(0.00 0.16 179.36)	(0.05 8.22 161.88)	(0.02 4.79 169.50)
Stick flesh	(0.00 0.16 179.57)	(0.09 8.93 155.07)	(0.14 12.81 148.82)
Tree	(0.00 0.05 179.82)	(0.22 17.13 140.18)	(0.22 17.21 140.45)
算法 C	算法 D	算法 E	本文方法
(0.00 2.11 175.61)	(0.17 6.12 145.58)	(0.15 9.13 147.14)	(0.11 7.15 137.42)
(0.03 6.70 166.57)	(0.13 10.10 148.55)	(0.23 14.27 139.74)	(0.24 8.48 133.95)
(0.02 5.51 168.20)	(0.22 9.76 140.13)	(0.32 14.11 131.87)	(0.31 13.77 130.97)
(0.01 4.04 169.87)	(0.25 14.53 134.61)	(0.33 14.71 129.68)	(0.22 8.59 128.26)
(0.15 11.88 147.81)	(0.25 17.90 136.74)	(0.32 18.32 131.45)	(0.30 12.28 128.45)
(0.00 1.79 176.23)	(0.11 10.52 152.62)	(0.10 10.78 153.62)	(0.28 12.06 134.22)
(0.02 4.58 167.23)	(0.17 9.68 145.50)	(0.24 12.81 135.21)	(0.21 10.93 133.57)
(0.05 8.64 162.26)	(0.37 16.70 127.05)	(0.35 18.70 127.05)	(0.30 15.28 125.96)

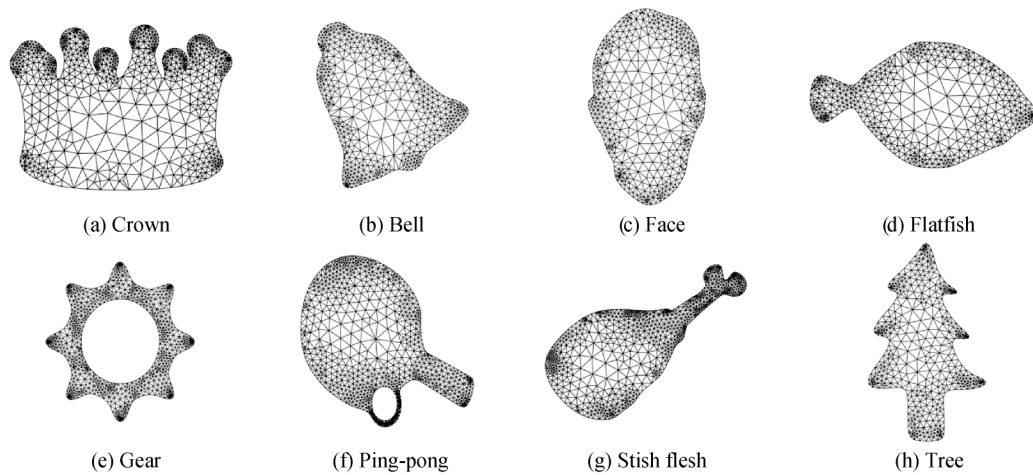


图 9 二值图像生成网格本文算法优化后的图像

本文算法求解的是一个非线性优化问题,其最为耗时部分为每次迭代所求解的线性方程组。表 4 列出了图 9 各个二维网格采用不同算法进行优化所

消耗的时间。由表 4 可见,本文算法的运行效率与算法 B、C、D 大致相当。因此,在不牺牲算法效率的同时,本文能够获得相对较好的结果。

表 4 网格规模以及各算法运行时间

网格名称	网格规模 (顶点数 \times 面数)	运行时间/ms					
		算法 A	算法 B	算法 C	算法 D	算法 E	本文方法
Crown	992 \times 1782	2	1137	1193	1360	64	472
Bell	567 \times 1017	1	669	774	581	36	755
Face	507 \times 909	1	584	728	462	30	258
Flatfish	527 \times 945	1	611	1025	513	35	715
Gear	1000 \times 1800	2	1141	1159	711	53	342
Ping-pong	1000 \times 1800	2	1149	1174	951	57	1394
Stick flesh	992 \times 1782	2	1155	1141	1284	59	1369
Tree	512 \times 918	<1	585	636	421	32	670

3 结 论

本文提出了一种基于角度的平面网格优化方法。该方法设计与网格拓扑连接相关的理想角度作为优化目标,在角度空间构造全局相容性约束,并采用序列线性约束规划法进行全局求解带约束的优化问题,最后基于所求角度运用最小二乘共形映射重建网格,并采用局部优化法进一步改善最小角度量。实验结果表明:本文提出的网格优化法与网格的坐标初值无关;其最大角度量具有显著优势,在大多数情况下均优于已有方法,且在平均意义上能够获得形状更好的三角形。但是该方法对最小角的优化效果不够明显,同时由于全局相容约束的非线性程度较高,数值优化的难度较大,算法的求解效率不够理想。在未来的工作中,将采用更为有效的网格度量函数作为优化目标使其能提高最小角,并将考虑采用增广拉格朗日法等数值优化方法进行改善,同时尝试将该方法推广至其他类型的网格。

参考文献:

[1] Botsch M, Kobbelt L, Pauly M, et al. Polygon Mesh Processing[M]. London: A K Peters/CRC Press, 2010: 1-20.

[2] Cheng S W, Dey T K, Shewchuk J R. Delaunay Mesh Generation[M]. Boca Raton: CRC Press, 2012: 31-56.

[3] Field D A. Laplacian smoothing and Delaunay triangulations [J]. Communications in Applied Numerical Methods, 1988, 4(6): 709-712.

[4] Brewer M L, Diachin L F, Knupp P M, et al. The mesquite mesh quality improvement toolkit [C]//Proceedings, 12th International Meshing Roundtable, Sandia National Laboratories, Santa Fe, New Mexico: ACAPS, 2003: 239-250.

[5] 李毅, 鲍劲松, 金烨, 等. 有限元平面混合网格优化策略[J]. 机械设计与研究, 2008(3): 6-9.

[6] Peng W, Ji W X. Surface mesh quality improvement by weighted inscribed circle method [J]. Advances in Engineering Software, 2019, 136: 102693.

[7] Yang F, Zhang D J, Ren H, et al. 2D mesh smoothing based on Markov chain method[J]. Engineering with Computers, 2020, 36(4): 1615-1626.

[8] Mittal K, Fischer P. Mesh smoothing for the spectral element method[J]. Journal of Scientific Computing, 2019, 78(2): 1152-1173.

[9] Knupp P M, Robidoux N. A framework for variational grid generation: conditioning the Jacobian matrix with matrix norms [J]. SIAM Journal on Scientific Computing, 2000, 21(6): 2029-2047.

[10] Munson T. Mesh shape-quality optimization using the inverse mean-ratio metric [J]. Mathematical Programming, 2007, 110(3): 561-590.

[11] Jiao X M, Wang D, Zha H Y. Simple and effective variational optimization of surface and volume triangulations [C]//Proceedings of the 17th International Meshing Roundtable. Berlin, Heidelberg: Springer, 2008: 315-332.

[12] Lipman Y. Bounded distortion mapping spaces for triangular meshes[J]. ACM Transactions on Graphics, 2012, 31(4): 108.

[13] Fu X M, Liu Y, Guo B N. Computing locally injective mappings by advanced MIPS[J]. ACM Transactions on Graphics, 2015, 34(4): 71.

[14] Durand R, Pantoja-Rosero B G, Oliveira V. A general mesh smoothing method for finite elements[J]. Finite Elements in Analysis and Design, 2019, 158: 17-30.

[15] Zhou T, Shimada K. An angle-based approach to two-dimensional mesh smoothing[J]. Proceedings of the 9th International Meshing Roundtable, 2000 (412): 373-384.

[16] Xu H T, Newman T S. An angle-based optimization approach for 2D finite element mesh smoothing[J]. Finite Elements in Analysis and Design, 2006, 42

- (13): 1150-1164.
- [17] 李宗领, 阎春平. 平面区域有限元三角网格迭代优化方法[J]. 新型工业化, 2013, 3(8): 32-40.
- [18] Renka R J. Mesh improvement by minimizing a weighted sum of squared element volumes [J]. International Journal for Numerical Methods in Engineering, 2015, 101(11): 870-886.
- [19] Renka R J. Two simple methods for improving a triangle mesh surface[J]. Computer Graphics Forum, 2016, 35(6): 46-58.
- [20] Xu K J, Gao X F, Chen G N. Hexahedral mesh quality improvement via edge-angle optimization[J]. Computers & Graphics, 2018, 70: 17-27.
- [21] Sheffer A, Lévy B, Mogilnitsky M, et al. ABF++: fast and robust angle based flattening [J]. ACM Transactions on Graphics, 2005, 24(2): 311-330.
- [22] Sheffer A, de Sturler E. Parameterization of faceted surfaces for meshing using angle-based flattening[J]. Engineering with Computers, 2001, 17(3): 326-337.
- [23] Lévy B, Petitjean S, Ray N, et al. Least squares conformal maps for automatic texture atlas generation[J]. ACM Transactions on Graphics, 2002, 21(3): 362-371.
- [24] Paul Chew L. Constrained Delaunay triangulations[J]. Algorithmica, 1989, 4(1): 97-108.

(责任编辑:康 锋)