博 士 学 位 论 文

面向等几何分析的区域参数化方法研究

On Domain Parameterization for Isogeometric Analysis

作 者 姓 名: 纪 野

学 号: 11901014

指 导 教 师: 朱 春 钢 教授

学科、 专业: 计 算 数 学 专业

答 辩 日 期: 2023 年 11 月 21 日

大连理工大学

Dalian University of Technology

学位论文原创性申明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经指明引用的内容外，学位论文不含任何其他个人、法人或者非法人组织已经发表或尚未发表的作品，且学位论文中已经指明作者姓名或者名称、作品名称的内容，不影响该作品的正常使用，也不存在不合理地损害相关权利人的合法权益的任何情形。对学位论文研究做出重要贡献的个人和法人或者非法人组织，均已在论文中以明确方式标明，且不存在任何著作权纠纷。

若因声明不实，本人愿意为此承当相应的法律责任。

学位论文题目：_____面向等几何分析的区域参数化方法研究_____

作 者 签 名 ： _____ 日期：_____ 年 ____ 月 ____ 日

大连理工大学学位论文版权使用授权书

本人完全了解大连理工大学有关学位论文知识产权的规定，在校攻读学位期间论文工作的知识产权属于大连理工大学，允许论文被查阅和借阅。学校有权保留论文并向国家有关部门或机构送交论文的复印件和电子版，可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印、或扫描等复制手段保存和汇编本学位论文。

学位论文题目：_____面向等几何分析的区域参数化方法研究_____

作 者 签 名 ： _____ 日期：_____ 年 ____ 月 ____ 日

导 师 签 名 ： _____ 日期：_____ 年 ____ 月 ____ 日

摘要

现代产品设计迭代流程依赖于精准且快速的物理仿真分析, 这使得计算机辅助设计 (CAD) 与计算机辅助工程 (CAE) 的无缝融合成为关键. 为了实现这一目标, 等几何分析 (IGA) 作为一种新型数值方法应运而生, 其核心思想是采用与表示 CAD 模型相同的样条基函数逼近物理场. 在基于 IGA 的全流程一体化设计–分析–优化中, *如何从 CAD 模型的边界表示出发构建适合分析的高质量区域参数化*是一个至关重要的问题, 其质量直接影响后续物理仿真分析的准确性和计算效率. 本论文聚焦于这一问题, 旨在提出构造高质量且适用于复杂物理域的区域参数化方法.

本论文不仅研究具有良好正交性和均匀性的各向同性参数化方法, 还研究更适合具有局部特征问题的 r 自适应参数化方法, 即各向异性参数化方法. 这种各向异性参数化方法能够更好地捕捉区域内物理问题的特征, 有助于提高分析的精度和效率. 本文核心代码在 Geometry + Simulation Modules (`https://github.com/gismo/gismo/`) 中开源, 其余主要贡献如下:

1. 对于一般的 NURBS 参数化, 本文提出了一系列参数化双射的充分条件与必要条件. 为提高计算域参数化的质量, 引入了内部控制顶点和权因子交替优化策略. 实现了基于障碍函数的无约束优化方法, 该方法避免了在迭代过程中计算复杂且耗时的约束条件. 不仅显著提升了参数化的质量, 还有效地提高了计算效率.
2. 为克服基于障碍函数参数化方法的局限性, 提出了一种基于罚函数的体参数化方法. 通过散度定理计算物理域的体积, 构造了刻画参数化体积畸变的目标泛函项. 此外, 本文提出了一种新的惩罚函数, 有效降低了计算误差. 结合雅可比正则化思想与减缩数值积分策略, 有效提升了新方法的鲁棒性与计算效率.
3. 对于具有极端纵横比的狭长计算域, 传统参数化技术存在数值困难, 收敛缓慢. 为此, 本文提出了一种改进的基于椭圆偏微分方程的参数化方法. 通过引入雅克比放缩因子, 显著提升了参数化质量. 为进一步提升数值稳定性与计算效率, 推导了对应的解析雅可比矩阵. 为提高所涉及非线性方程组的求解效率, 提出了预条件 Anderson 加速求解器, 同时给出了该方法的残差估计和收敛性分析.
4. 在应用层面, 成功将所提出的参数化方法应用到双螺杆压缩机的网格生成问题中. 利用不同的网格离散方案, 能够轻易地生成具有边界层特性的网格, 还能实现与流场方向一致的网格布局. 在商用软件 ANSYS CFX ™ 中的测试表明, 本方法生成的网格可以适用于复杂的流体动力学仿真分析.
5. 针对具有局部特征的复杂物理问题, 本文提出了一种基于曲率度量的各向异性参数化方法. 将 IGA 数值解视为参数曲面, 借助绝对主曲率表征 IGA 数值解的变化情况, 建立了几何量与分析结果之间的紧密联系. 为提高基于梯度优化算法

的效率和稳定性, 推导了本方法的灵敏度传播和解析灵敏度计算公式. 采用双层优化策略, 实现在保持计算精度的同时提高计算效率, 有效降低了计算复杂度.

综上所述, 本论文提出了几类面向等几何分析的高质量区域参数化方法, 这些方法在计算效率、数值精度和鲁棒性方面具有优势, 为解决复杂几何外形计算域上的物理问题提供了理论基础和快速解决方案, 从而为基于等几何分析的 CAD/CAE 无缝融合提供了重要的技术支持. 此外, 这些方法在实际工程应用中具有潜在价值, 为工程师和设计师提供了可靠且高效的方法和工具.

关键词: 等几何分析, 适合分析的参数化, 各向同性/异性参数化, 自适应, NURBS

ABSTRACT

In modern product design iterations, precise and rapid physical simulation analyses are imperative, emphasising the necessity for seamless integration between Computer-Aided Design (CAD) and Computer-Aided Engineering (CAE). Isogeometric Analysis (IGA) has been introduced as an innovative numerical technique to fulfil this demand. Central to IGA is the utilization of spline basis functions identical to those employed in CAD models for approximating physical fields. Within an IGA-based design-analysis-optimization integrated framework, a pivotal challenge lies in *constructing high-quality domain parameterizations suitable for simulation, originating from the boundary representations of CAD models*. The parameterization quality is directly consequential to the accuracy and computational efficiency of the downstream physical simulations. This dissertation concentrates on this significant issue, aspiring to present effective methodologies for generating high-quality parameterizations that are adaptable to complex physical domains.

This dissertation not only investigates isotropic methods that exhibit commendable orthogonality and uniformity but also explores r-adaptive techniques tailored for problems with localized features, namely, anisotropic parameterization methods. The core code is open-sourced and available in the Geometry + Simulation Modules (`https://github.com/gismo/gismo/`). Additional key contributions are as follows:

1. This dissertation introduces both sufficient conditions and necessary conditions for the bijectivity of general NURBS parameterizations. We propose an alternating optimization strategy that focuses on interior control points and weights to enhance parameterization quality. Additionally, we employ a barrier function-based unconstrained optimization technique, which simplifies the computational process by removing challenging and time-consuming constraints. This strategy effectively improves computing efficiency while also improving parameterization quality.
2. We suggest a penalty function-based approach to address the limitations associated with the aforementioned barrier function-based domain parameterization method. The volume of the computational domain is computed using the divergence theorem, and an objective functional term that describes volume distortion is defined. In addition, we present a penalty function created to effectively reduce computational errors. The Jacobian regularization technique and reduced numerical integration schemes are integrated, which significantly improves the robustness and computational effectiveness.
3. Traditional domain parameterization techniques frequently encounter numerical chal-

lenges and sluggish convergence for computational domains with extreme aspect ratios. We provide an improved domain parameterization method based on elliptic partial differential equations to address these issues. The use of a Jacobian scaling factor considerably enhances the quality of domain parameterization. We derive the related analytical Jacobian matrix to improve numerical stability and computational effectiveness. In order to effectively solve the underlying nonlinear equation systems, we also develop a preconditioned Anderson acceleration solver. Residual bound estimates and a thorough convergence analysis accompany the solver.

4. Our suggested domain parameterization techniques have been effectively used in real-world twin-screw compressor mesh-generating applications. We enable the seamless construction of grids that capture boundary-layer properties and align with flow fields using multiple mesh discretization algorithms. The grids produced by our approaches are suitable for sophisticated fluid dynamics simulations, according to validation tests carried out in the commercial software ANSYS CFX ™.
5. For complex physical problems with localized features, we propose a curvature-based anisotropic domain parameterization technique for challenging physical issues with localized features. In this approach, IGA solutions are interpreted as parameter surfaces, and their absolute principal curvatures are used to describe variations of the IGA solutions. This creates a subtle connection between geometric elements and analytical results. We create both analytical sensitivity formulas and sensitivity propagation methods to improve the stability and effectiveness of gradient-based optimization algorithms. For the purpose of balancing computational precision and efficiency, a bi-level optimization technique is employed, which significantly lowers the amount of computing power involved in the domain parameterization procedure.

In conclusion, this dissertation offers a variety of advanced domain parameterization techniques specifically developed for isogeometric analysis. The methods introduced represent significant improvements in computational efficiency, numerical accuracy, and robustness. They provide significant technical assistance for the smooth integration of CAD/CAE based on isogeometric analysis since they offer both a theoretical foundation and quick solutions for handling physical issues in complicated geometric computational domains. Furthermore, these techniques give engineers and designers dependable and effective tools and approaches that may be used in real-world engineering applications.

Key Words: Isogeometric analysis, Analysis-suitable parameterization, Isotropic / Anisotropic parameterization, Adaptivity, NURBS

目　　录

摘要 I
ABSTRACT III
TABLE OF CONTENTS IX
图目录 XII
表目录 XV
主要符号表 XIX
1 绪论 1
1.1 研究背景与意义 1
1.2 国内外相关研究工作进展 5
1.2.1 适合分析的各向同性参数化方法 5
1.2.2 适合分析的各向异性参数化方法 8
1.3 本文主要内容与贡献 8
1.4 本文结构安排 10
2 预备知识 13
2.1 非均匀有理 B 样条 13
2.2 等几何分析简述: 以 Poisson 方程为例 17
2.3 适合分析的参数化 19
2.3.1 适合分析参数化问题的提出 20
2.3.2 适合分析参数化存在的前提条件 21
2.3.3 各向同性参数化质量指标 22
3 基于障碍函数的高质量平面参数化方法 25
3.1 引言 25
3.2 单值性条件和检验算法 26
3.2.1 NURBS 参数化单值性的充分条件 26
3.2.2 弱化的单值性条件 28
3.2.3 单值性检验算法 30
3.3 基于障碍函数的参数化方法与交替更新策略 32
3.3.1 第一步: 初始参数化构造 32
3.3.2 第二步: 消除折叠 33
3.3.3 第三步: 参数化质量提升 35

3.4 数值实验与对比分析 37
3.4.1 算法实现细节 38
3.4.2 消除折叠的必要性 39
3.4.3 权因子的作用 40
3.4.4 与现有方法的对比分析 42
3.4.5 计算效率对比 45
3.5 本章小结 46
4 基于罚函数的高质量体参数化方法 47
4.1 引言 47
4.2 罚函数与雅可比正则化技术 48
4.2.1 初始参数化构造 49
4.2.2 加权目标泛函 49
4.2.3 雅可比正则化技术 51
4.3 算法效率上的几点考虑 53
4.3.1 灵敏度分析 53
4.3.2 高效的减缩积分方案 54
4.3.3 数值优化算法 54
4.3.4 基于罚函数的参数化算法总览 55
4.4 数值实验与对比分析 55
4.4.1 算法实现细节 56
4.4.2 初始参数化对参数化结果的影响 57
4.4.3 鲁棒性测试 58
4.4.4 与高精度 Gauss 积分方案对比 60
4.4.5 平衡参数 λ_{mips} 和 λ_{vol} 对参数化质量的影响 62
4.4.6 与现有方法的对比分析 62
4.5 在等几何分析中的应用 66
4.6 本章小结 67
5 改进的椭圆参数化方法与预条件 Anderson 加速算法 68
5.1 引言 68
5.2 椭圆参数化构造方法 70
5.2.1 Sobolev 空间 H^2 中的离散格式 71
5.2.2 H^2 离散方法的解析雅可比计算 71
5.2.3 Sobolev 空间 H^1 中的缩放离散格式 72
5.2.4 缩放 H^1 离散的解析雅可比计算 74

5.2.5 非线性方程组的等价固定点迭代问题 75
5.3 预条件 Anderson 加速算法 75
5.3.1 非线性预条件子 75
5.3.2 算法收敛性分析 77
5.4 数值实验与对比分析 80
5.4.1 算法实现细节 80
5.4.2 不同非线性求解算法的性能比较 81
5.4.3 缩放 H^1 方法对参数化质量的影响 83
5.4.4 与现有方法的对比分析 84
5.5 在等几何分析中的应用 86
5.6 各向同性参数化方法比较 89
5.7 本章小结 91
6 基于样条参数化的双螺杆压缩机网格生成方法 93
6.1 引言 93
6.2 主要工作流程 94
6.3 参数化方法 95
6.3.1 预处理步骤 95
6.3.2 基 “O” 形参数化生成 98
6.4 狭窄间隙的参数化构造 99
6.5 网格离散化技术 100
6.5.1 边界层网格 101
6.5.2 流对齐网格 101
6.5.3 流体仿真分析 103
6.6 本章小结 103
7 基于绝对主曲率度量和双层优化的 r 自适应方法 104
7.1 引言 104
7.2 基于绝对主曲率度量的 r 自适应方法 105
7.2.1 绝对主曲率 105
7.2.2 目标泛函 107
7.3 灵敏度分析 108
7.3.1 灵敏度传播: 从设计模型到分析模型 108
7.3.2 灵敏度分析 110
7.3.3 伪载荷的解析计算 113

7.4 数值实验与对比分析 115
7.4.1 算法实现细节 115
7.4.2 与现有方法对比的设定 116
7.4.3 一个简单例子: 单位正方形 116
7.4.4 NURBS 几何: 四分之一圆环 119
7.4.5 更复杂的计算域几何 121
7.4.6 加速策略 122
7.5 在带有移动 Guassian 热源热传导问题上的应用 123
7.6 本章小结 126
8 总结与展望 127
8.1 本文工作总结 127
8.2 未来工作展望 128
8.3 创新点摘要 130
参考文献 131
攻读博士学位期间科研项目及科研成果 141
致谢 145
作者简介 149

TABLE OF CONTENTS

1 Introduction 1
1.1 Research Background and Significance 1
1.2 Domestic and International Research Progress 5
1.2.1 Review: Analysis-Suitable Isotropic Parameterization Methods 5
1.2.2 Review: Analysis-Suitable Anisotropic Parameterization Methods 8
1.3 Main Content and Contributions 8
1.4 Structural Organization 10
2 Preliminaries 13
2.1 Non-Uniform Rational B-Splines 13
2.2 Brief Overview of Isogeometric Analysis: Poisson Equation 17
2.3 Analysis-suitable Parameterization 19
2.3.1 Problem Statement for Analysis-Suitable Parameterization 20
2.3.2 Existence Prerequisites for Analysis-Suitable Parameterization 21
2.3.3 Quality Metrics for Isotropic Parameterization 22
3 Barrier Function-Based High-Quality Planar Parameterization Method 25
3.1 Introduction 25
3.2 Bijectivity Conditions and Verification Algorithm 26
3.2.1 Sufficient Conditions for Bijectivity of NURBS Parameterization 26
3.2.2 Weaker Bijectivity Conditions 28
3.2.3 Bijectivity Verification Algorithm 30
3.3 Barrier Function-Based Method and Alternating Update Strategy 32
3.3.1 Step I: Initial Parameterization Construction 32
3.3.2 Step II: Foldover Elimination 33
3.3.3 Step III: Parameterization Quality Improvement 35
3.4 Numerical Experiments and Comparisons 37
3.4.1 Implementation Details 38
3.4.2 Necessity of Foldover Elimination 39
3.4.3 Role of Weights 40
3.4.4 Comparative Analysis with Existing Methods 42
3.4.5 Computational Efficiency Comparison 45
3.5 Chapter Summary 46

4 Penalty Function-Based High-Quality Volumetric Parameterization Method 47
4.1 Introduction 47
4.2 Penalty Function and Jacobian Regularization Technique 48
4.2.1 Initial Parameterization Construction 49
4.2.2 Weighted Objective Functional 49
4.2.3 Jacobian Regularization Technique 51
4.3 Some Considerations on Algorithmic Efficiency 53
4.3.1 Sensitivity Analysis 53
4.3.2 Efficient Reduced Quadrature Scheme 54
4.3.3 Numerical Optimization Algorithm 54
4.3.4 Overview of Penalty Function-Based Parameterization Algorithm 55
4.4 Numerical Experiments and Comparisons 55
4.4.1 Implementation Details 56
4.4.2 Effects of Initialization 57
4.4.3 Robustness Testing 58
4.4.4 Comparison with High-Precision Gaussian Quadrature Scheme 60
4.4.6 Comparative Analysis with Existing Methods 62
4.5 Applications in Isogeometric Analysis 66
4.6 Chapter Summary 67
5 Improved Elliptic Parameterization Method and Preconditioned Anderson Acceleration Algorithm 68
5.1 Introduction 68
5.2 Elliptic Parameterization Method 70
5.2.1 Discretization in Sobolev Space H^2 71
5.2.2 Analytical Jacobian Calculation for H^2 Discretization 71
5.2.3 Scaled Discretization in Sobolev Space H^1 72
5.2.4 Analytical Jacobian Calculation for Scaled H^1 Discretization 74
5.2.5 Equivalence of Nonlinear System and Fixed-Point Iteration Problem 75
5.3 Preconditioned Anderson Acceleration Algorithm 75
5.3.1 Nonlinear Preconditioner 75
5.3.2 Convergence Analysis 77
5.4 Numerical Experiments and Comparisons 80
5.4.1 Implementation Details 80
5.4.2 Performance Comparison of Different Nonlinear Solvers 81

5.4.3 Role of Scaled H^1 Method on Parameterization Quality 83
5.4.4 Comparative Analysis with Existing Methods 84
5.5 Applications in Isogeometric Analysis 86
5.6 Comparison of Isotropic Parameterization Methods 89
5.7 Chapter Summary 91
6 Mesh Generation for Twin-Screw Compressors by Spline-based Parameterization 93
6.1 Introduction 93
6.2 Main Workflow 94
6.3 Parameterization Methodology 95
6.3.1 Pre-processing Steps 95
6.3.2 Base O-type Parameterization Generation 98
6.4 Parameterization Construction for Narrow Clearance 99
6.5 Mesh Discretization Techniques 100
6.5.1 Boundary-Layer Mesh 101
6.5.2 Flow-aligned Mesh 101
6.5.3 Fluid Simulation Analysis 103
6.6 Chapter Summary 103
7 Curvature-based r-Adaptive Method Using Bi-Level Approach 104
7.1 Introduction 104
7.2 r-Adaptive Method Based on Absolute Principal Curvature Metrics 105
7.2.1 Absolute Principal Curvature 105
7.2.2 Objective Functional 107
7.3 Sensitivity Analysis 108
7.3.1 Sensitivity Propagation: From Design Model to Analysis Model 108
7.3.2 Sensitivity Analysis 110
7.3.3 Analytical Calculation of Pseudo-Loads 113
7.4 Numerical Experiments and Comparisons 115
7.4.1 Implementation Details 115
7.4.2 Comparison with Existing Methods Settings 116
7.4.3 A Toy Problem: Unit Square 116
7.4.4 NURBS Geometry: Quarter Annulus 119
7.4.5 More Complicated Computational Domain Geometry 121
7.4.6 Acceleration Strategy 122

7.5 Application to Heat Transfer Problem with a Moving Gaussian Heat Source..... 123
7.6 Chapter Summary..... 126
8 Conclusions and Future Outlooks 127
8.1 Conclusions 127
8.2 Future Work Outlooks 128
8.3 Innovation Summaries..... 130
References..... 131
Achievements..... 141
Acknowledgements..... 145
CV 149

图 目 录

图 1.1 Utah 茶壶、茶杯和勺子: 分别由 32、26 和 16 片 Bézier 曲面片拼接而成 1
图 1.2 有限元分析与离散网格 2
图 1.3 一段管道的设计-分析-结构优化流程 3
图 1.4 各向同性参数化 (左) 和各向异性参数化 (右) 5
图 1.5 自然界中广泛存在局部和多尺度特征 9
图 1.6 本文主要内容和结构安排 11
图 2.1 开节点区间 $\Xi^* = \{0,0,0,0.2,0.4,0.6,0.8,0.8,1,1,1\}$ 上的二次 B 样条基函数和曲线 14
图 2.2 双二次张量积 B 样条基函数和曲面 15
图 2.3 一个 $2\times2\times2$ 次张量积 B 样条体 16
图 2.4 $\mathbb{R}^2$ 中圆的 NURBS 表示由 $\mathbb{R}^3$ 中的二次 B 样条曲线经中心投影变换得到 17
图 2.5 等几何分析计算示意图 18
图 2.6 适合分析参数化问题的示意图 20
图 2.7 不可能由单块 C^k $(k\geq1)$ 单值 NURBS 体参数化的计算域 21
图 2.8 雅可比行列式的几何意义 22
图 2.9 雅可比矩阵奇异值的几何意义 23
图 3.1 区域参数化: 从边界表示构造参数化 $\boldsymbol{x}$ 25
图 3.2 初始化和消除折叠 33
图 3.3 求解无约束问题 (3.12) 得到几乎单值的参数化 34
图 3.4 有/无均匀性泛函项的参数化结果对比 37
图 3.5 消除折叠 + 修正 Winslow 泛函 + 均匀泛函方法参数化结果及质量指标 39
图 3.6 固定权因子方法与交替更新控制顶点和权因子方法参数化结果对比 40
图 3.7 飞机: 不同参数化方法得到的参数化结果 42
图 3.8 海豚: 不同参数化方法得到的参数化结果 43
图 3.9 飞机: 比例雅可比 $|\mathcal{J}|_s$(第一行) 和条件数 $\kappa(\mathcal{J})$(第二行) 43
图 3.10 海豚: 比例雅可比 $|\mathcal{J}|_s$(第一行) 和条件数 $\kappa(\mathcal{J})$(第二行) 44
图 3.11 狗: 低秩拟共形映射方法[1] 和本章方法生成的参数化结果与质量比较 44
图 3.12 企鹅: 低秩拟共形映射方法[1] 和本章方法生成的参数化结果与质量比较 45
图 4.1 体参数化问题: 构造从参数域 $\hat{\Omega}$ 到计算域 Ω 的单值样条映射 $\boldsymbol{x}$ 47
图 4.2 前向 Laplacian 方法得到的初始参数化 49
图 4.3 惩罚函数与雅可比正则化技术 52
图 4.4 平面区域上双三次 NURBS 参数化减缩积分方案 54

图 4.5 Rot90: 由不同初始参数化得到的参数化结果 57
图 4.6 鸭子: 由不同初始参数化得到的参数化结果 58
图 4.7 分别扭转一个长方体 180°, 360°, 540°, and 720° 得到的参数化结果 59
图 4.8 六个复杂模型的边界表示及其几何信息 (p,q,r,n,m,l) 60
图 4.9 减缩积分方案与高精度数值积分方案的参数化结果对比 61
图 4.10 不同参数化方法的计算时间 (秒) 61
图 4.11 不同平衡参数 λ_{mips} 和 λ_{vol} 对参数化结果的影响 62
图 4.12 鸭子: 与潘茂东等[2] 和刘浩等[3] 方法的对比 63
图 4.13 猴子: 与潘茂东等[2] 和刘浩等[3] 方法的对比 63
图 4.14 本章方法构造的体参数化上求解 Poisson 方程 65
图 4.15 不同方法构造的体参数化上求解 Poisson 方程的绝对误差分布 66
图 5.1 螺杆机械截面: 罚函数方法和椭圆网格生成方法的参数化结果 68
图 5.2 鸭子: H^2 离散参数化结果 (左) 和缩放 H^1 离散参数化结果 (右) 73
图 5.3 阳转子: 不同非线性求解算法的数值表现对比 81
图 5.4 蝴蝶: 不同非线性求解算法的数值表现对比 82
图 5.5 不同非线性求解算法在四片参数化的双螺杆几何上的表现 83
图 5.6 飞机: H^2 离散和缩放的 H^1 离散生成的参数化结果 84
图 5.7 螺杆与机匣: H^2 离散和缩放的 H^1 离散生成的参数化结果 86
图 5.8 平面阳转子几何: 边界条件、精确解及绝对误差分布 87
图 5.9 h 细化过程中误差收敛和刚度矩阵 Frobenius 条件数变化曲线 88
图 5.10 三维转子几何: 等几何分析数值解和绝对误差分布 (自由度为 $30,704$) 88
图 5.11 h 细化过程中误差收敛曲线 88
图 5.12 977 平面数据集中部分模型 89
图 5.13 不同方法的最差质量指标: $\min\left(|\mathcal{J}|_s\right)$ vs. $\max\left(unif.\right)$ 90
图 5.14 不同方法的平均质量指标: $\min\left(|\mathcal{J}|_s\right)$ vs. $\max\left(unif.\right)$ 91
图 5.15 不同方法的计算时间 91
图 6.1 双螺杆旋转压缩机的基本结构[1] 93
图 6.2 本章方法的主要工作流程 95
图 6.3 外边界正则化方法和阳转子的参数匹配结果 96
图 6.4 阴转子: 两种椭圆参数化方法得到的结果 98
图 6.5 分割基 “O” 形参数化以获得外部表示和狭窄间隙 98
图 6.6 边界参数匹配方法和参数匹配前后的参数化结果 99
图 6.7 转子旋转时平面参数化的演变 100
图 6.8 沿 ζ-参数方向的一族平面参数化和体参数化结果 100

图 6.9 利用参数变换生成具有边界层特性的结构六面体网格 101
图 6.10 利用参数变换生成流对齐的网格 102
图 6.11 不同类型网格在 ANSYS CFX ™ 中的仿真结果.....103
图 7.1 双层优化 r 自适应参数化流程示意图.....105
图 7.2 等几何分析解曲面, 数值误差及绝对主曲率分布 107
图 7.3 单位正方形: 不同细化层次分析模型对数值误差的影响.....117
图 7.4 单位正方形: h 细化过程中误差收敛曲线.....118
图 7.5 四分之一圆环: 输入模型和精确解 120
图 7.6 四分之一圆环: 不同方法间的比较 120
图 7.7 鸭子: 设计模型和精确解 121
图 7.8 加速策略: 仅将内部红色的控制顶点作为优化变量 122
图 7.9 移动热源问题描述: 热源沿着圆弧移动, 红色箭头表示扫描路径 123
图 7.10 不同时刻的解和参数化结果 125
图 7.11 不同时刻 t 的相对总能量误差 $\varepsilon(\mathscr{U}_{(t)}, \mathscr{U}_{(t)})$, 此时自由度为 1225.....126
图 8.1 六面体块分割 128

表 目 录

表 3.1 消除折叠的必要性: 不同方法所构造参数化结果的有效性 39
表 3.2 不同方法所构造参数化结果的量化统计数据 41
表 3.3 不同参数化方法的计算时间 (秒) 46
表 4.1 不同初始参数化得到的参数化结果的量化统计数据, 时间单位 (秒) 57
表 4.2 将长方体扭转不同角度得到的体参数化结果的质量统计. 59
表 4.3 考拉模型上不同平衡参数 λ_{mips} 和 λ_{vol} 组合得到的参数化质量统计 62
表 4.4 不同方法所构造参数化结果的量化统计数据 64
表 5.1 不同方法所构造参数化结果的量化统计数据 85
表 5.2 不同参数化方法的成功率 90
表 7.1 不同 r 自适应参数化方法的量化统计数据 119
表 7.2 鸭子模型上不同方法的量化统计数据 121

主要符号表

符号	代表意义	对应英文术语
$\mathbb{R}^d$	d 维欧式空间	d-dimensional Euclidean space
H^1, H^2	索布列夫空间	Sobolev space
$\hat{\Omega}$	参数域	Parameter Domain
Ω	计算域/物理域	Computational / Physical Domain
$\boldsymbol{\xi} = (\xi, \eta, \zeta)^{\mathrm{T}}$	参数坐标	Parameter Coordinate
$\boldsymbol{x} = (x, y, z)^{\mathrm{T}}$	物理坐标 / 参数化	Physical Coordinate / Parameterization
p, q, r	次数	Degree
$\Xi, \boldsymbol{\mathcal{H}}, \boldsymbol{\mathcal{Z}}$	节点向量	Knot Vector
$\mathbf{P}_i$	控制顶点	Control Point
ω_i	权因子	Weight
$N_i(\boldsymbol{\xi})$	B 样条基函数	B-spline Basis Function
$R_i(\boldsymbol{\xi})$	NURBS 基函数	NURBS Basis Function
$\mathscr{U}$	物理解	Physical Solution
$\mathscr{U}^{\text{exact}}$	精确解	Exact Solution
$\boldsymbol{\mathcal{J}}$	参数化的雅可比矩阵	Jacobian Matrix of Parameterization
$\lvert\boldsymbol{\mathcal{J}}\rvert$	参数化的雅可比行列式	Jacobian Determinant of Parameterization
$\lvert\boldsymbol{\mathcal{J}}\rvert_s$	比例雅可比	Scaled Jacobian
$\lVert\cdot\rVert_F$	Frobenius 范数	Frobenius Norm
$\mathrm{cond}(\cdot)$	Frobenius 条件数	Frobenius Condition Number
$m^{\text{unif.}}$	均匀性指标	Uniformity Metric
$\mathscr{E}$	目标泛函	Objective Functional
∂_p	关于控制顶点的偏导数	Partial Derivative w.r.t. Control Points
$area(\Omega)$	物理域的面积	Area of Physical Domain
$vol(\Omega)$	物理域的体积	Volume of Physical Domain
$\lambda_{\text{mips}}, \lambda_{\text{vol}}$	平衡参数	Trade-off Parameters
$\mathbf{K}$	刚度矩阵	Stiffness Matrix
$\mathbf{M}$	质量矩阵	Mass Matrix
$\boldsymbol{\mathcal{M}}_k$	非线性预条件子	Nonlinear Preconditioner
$jac(\boldsymbol{\mathcal{F}})$	非线性方程组的雅可比矩阵	Jacobian Matrix of Nonlinear System

缩写	代表意义	对应英文术语
NURBS	非均匀有理 B 样条	Non-Uniform Rational B-Spline
IGA	等几何分析	Isogeometric Analysis
FEA	有限元分析	Finite Element Analysis
CFD	计算流体动力学	Computational Fluid Dynamics
THB 样条	截断层次 B 样条	Truncated Hierarchical B-splines
PHT 样条	层次 T 网格上的多项式样条	Polynomial splines over Hierarchical T-meshes
G+Smo	一款 C++ 编写的等几何分析库	Geometry + Simulation Modules

注：如文中对符号另有说明，以文中对应位置说明为准。

1 绪论

1.1 研究背景与意义

计算机辅助设计 (Computer-Aided Design, 简称 CAD)[4,5] 是一门研究计算机在几何建模和设计中的应用的学科. 其核心在于使用计算机技术来表达、设计和分析几何对象, 这些对象通常包括曲线、曲面和实体. 如今, CAD 技术已成为现代工程设计、建筑设计、制造业、航空航天、汽车设计和动画制作等诸多领域的关键技术手段.

非均匀有理 B 样条 (Non-Uniform Rational Basis Spline, 简称 NURBS) 是现代 CAD 系统的标准表示形式. 几乎所有主流的 CAD 软件和几何建模工具都支持 NURBS 曲线曲面表示, 从而保证不同软件之间的数据交换. 现代样条技术的发展一般被认为起源于 1946 年 I.J. Schoenberg 使用样条函数对数据进行平滑[6]. 不过直到 1959 年和 20 世纪 60 年代, 法国雪铁龙汽车公司的 Paul de Casteljau 和雷诺汽车公司的 Pierre Bézier 分别独立发明了 Bézier 曲线, 人们才逐渐意识到样条工具在几何造型领域的巨大潜力[7]. 图 1.1 展示了几何造型领域著名的 Utah 茶壶、茶杯和勺子, 它们分别由 32、26 和 16 片 Bézier 曲面片拼接而成. 随后, 样条工具得到了迅猛发展, 包括著名的 Cox-de Boor 递推公式和 NURBS 的一系列几何算法 (如节点插入和升降阶算法等)[8], 这些发展进一步推动了 CAD 的应用.

图 1.1 Utah 茶壶、茶杯和勺子[1]: 分别由 32、26 和 16 片 Bézier 曲面片拼接而成
Fig. 1.1 Utah Teapot, Teacup, and Spoon[1]: Composed of 32, 26, and 16 Bézier patches, respectively

以有限元分析 (Finite Element Analysis, 简称 FEA) 为代表的计算机辅助工程 (Computer-Aided Engineering, 简称 CAE) 是一种应用计算机技术辅助工程分析、模拟和优化设计的方法. 现代 FEA 的开端通常被认为是 1943 年 Courant 发表的一篇利用变分

[1]数据来源: `ftp://ftp.funet.fi/pub/sci/graphics/packages/objects/teaset.tar.Z`

原理和三角形单元解决振动问题的论文[9]. 这项研究首次提出了将复杂结构划分为小单元的思想, 为有限元分析的发展奠定了基础. 1956 年, J. Turner、R. Clough、H. Martin 和 L. Topp 在航空航天领域开展研究, 首次使用 “有限元” 这一术语. 随着计算机技术的迅速发展, FEA 经历了从简单线性问题到复杂非线性问题, 从单物理场到多物理场的发展, 而且在工程领域取得了十分广泛的应用[10].

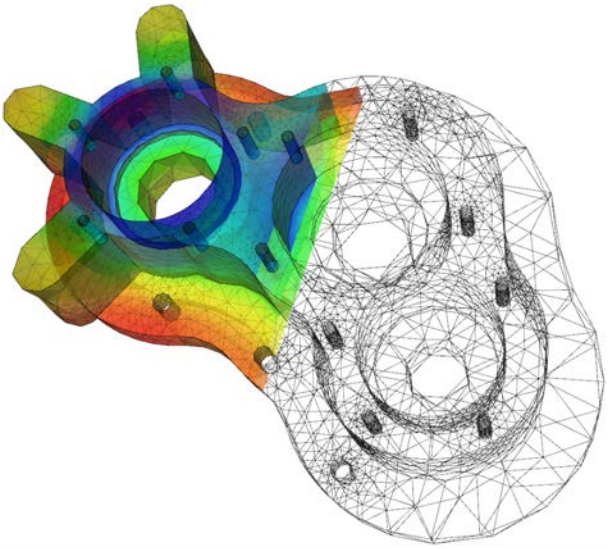


图 1.2　有限元分析与离散网格[1]
Fig. 1.2　Finite Element Analysis and Discrete Mesh[1]

虽然 CAD 和 CAE 都依赖计算机技术来解决工程问题, 但它们在历史上一直是两个平行发展的学科. CAD 主要关注几何建模和设计, 涉及创建和修改三维几何模型以满足工程、建筑和产品设计的需求. 而 CAE 则专注于产品设计的工程分析、模拟和优化, 以评估产品的性能、可靠性和有效性. 然而, 现代产品迭代循环通常要求在完成产品设计后, 对其进行模拟和分析, 以期提高产品的性能表现. 由于 CAD 软件中的产品模型通常采用基于 NURBS 的几何表示, 它们必须经过预处理才能用于 CAE 分析. 这一过程包括简化几何模型、消除几何错误、定义材料属性和施加边界条件等. 此外, CAE 软件通常需要将几何模型转换为离散的有限元网格, 如图 1.2 所示.

据美国 Sandia 国家实验室的数据显示[11], 在基于传统 FEA 的 “设计–分析–优化” 工作流程中, 几何处理和生成离散化网格占到整个工作流程中 80% 以上的时间, 并且需要大量的人工干预. 此外, 由于 CAE 中的网格模型仅仅是原 CAD 模型的近似, 在某些对几何误差非常敏感的问题上难以处理, 如接触问题和板壳屈曲问题等. 为了解决这些问题, 研究人员和工程师们需要继续改进现有的工具和方法, 以实现 CAD 与 CAE 之间更紧密的集成. 此外, CAD 和 CAE 软件之间的数据交换和兼容性被列为十

[1]图源:https://upload.wikimedia.org/wikipedia/commons/c/cd/Elmer-pump-heatequation.png

大 CAD 挑战之一[12]. 虽然许多 CAD 和 CAE 软件都支持通用的文件格式 (如 IGES, STEP 等), 但在实际操作中将 CAD 模型导入到 CAE 软件中进行分析时, 可能会遇到几何错误或单位不一致等问题, 从而导致分析结果不准确或分析失败. 据美国三角研究所 1999 年的报告估计[13], 仅美国汽车行业就因为设计和分析系统数据类型转换每年至少损失 10 亿美元, 并导致新型汽车的推出至少推迟两个月.

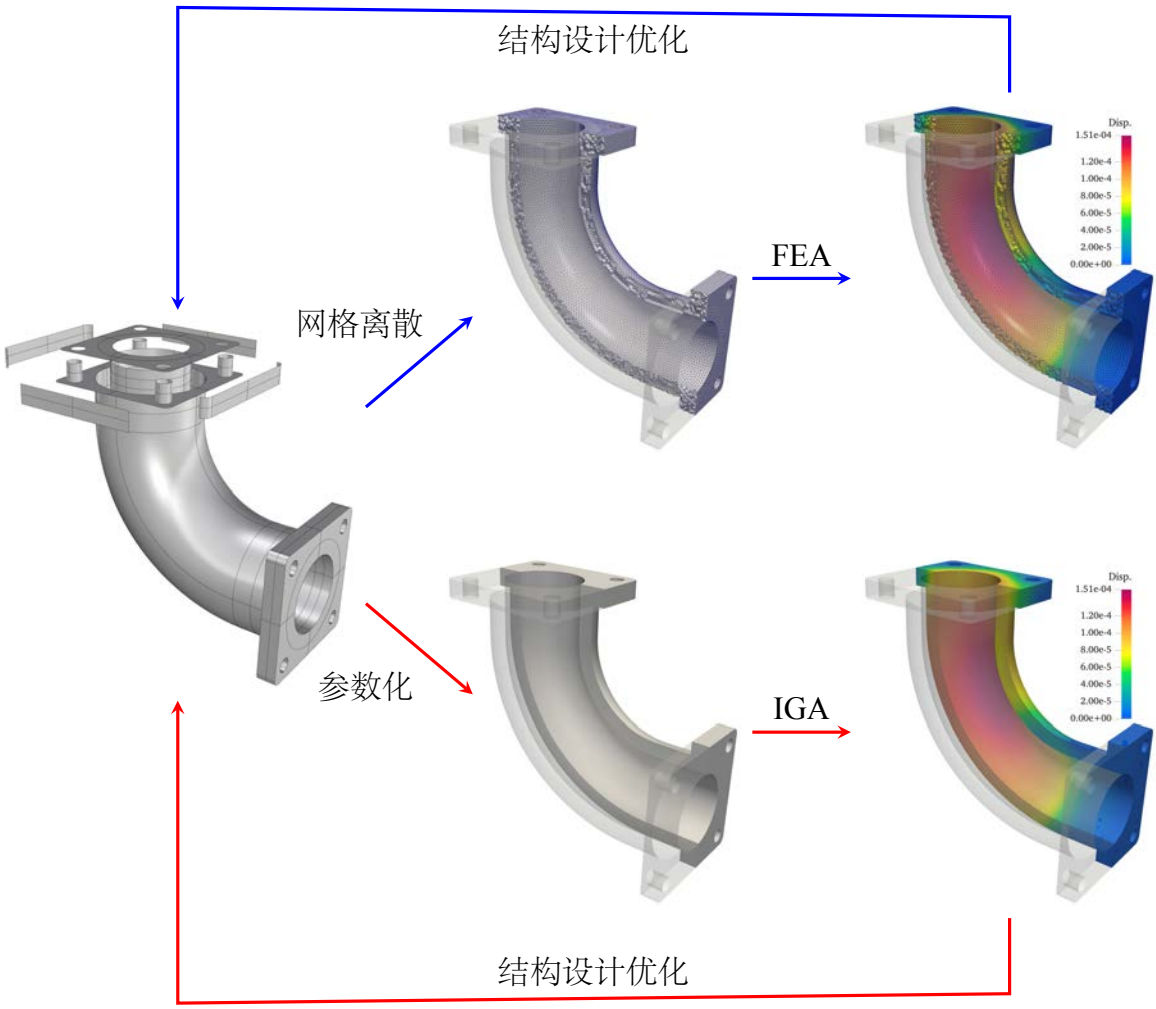


图 1.3　一段管道的设计-分析-结构优化流程
Fig. 1.3　Design-Analysis-Optimization Workflow for a Pipe Segment

在这一背景下, 等几何分析 (Isogeometric Analysis, IGA) 应运而生[11]. IGA 的概念最早由 Thomas J.R. Hughes 等人于 2005 年[14] 提出, 该方法将 CAD 设计中广泛使用的曲线、曲面和实体的表示方法 (如 NURBS) 直接应用到后续的分析中, 从而为 CAD 与 CAE 之间的无缝融合提供了新思路. 这一方法为具有复杂几何结构的 CAD 建模和 CAE 分析提供了一种更加高效且准确的方法. 因此, 在 IGA 提出后的十几年来, 逐渐受到了学术界和工程界的广泛关注, 如壳分析[15,16], 流体力学[17–19], 流固耦合[20,21], 电磁

学[22,23], 生物医学[24–26] 以及最近在增材制造领域火热的结构设计优化[27–33]. 此外, 一些工程领域所熟知的商业软件公司已经关注并投入研发 IGA 模块或插件, 如 Coreform IGA[1], ANSYS LS-DYNA[2], Rhino/Grasshopper[3]等.

相较于传统的有限元分析, 等几何分析在以下方面具有明显的优势:

1) 几何建模与分析的无缝衔接: 传统 FEA 需要对结构进行网格剖分, 而且在处理具有复杂几何的计算域或高度非线性问题时会出现网格失真的问题, 从而导致分析结果不准确. 与之相比, IGA 可以精确地表示 CAD 模型, 消除了几何误差并且避免了网格失真问题, 实现了几何建模与数值分析的无缝衔接. 另一方面, 这也极大地简化了设计与分析流程, 节省了大量时间和资源. 特别是对于复杂的三维结构, IGA 在提高工程设计与分析效率方面具有显著优势[11,14]. 图 1.3 展示了基于传统 FEA 和基于 IGA 的 “设计–分析–优化” 流程, FEA 需要离散网格, 而在基于 IGA 的方法中全流程采用样条表示.
2) 更高的逼近精度与光滑性: IGA 使用精确和高阶光滑的曲线和曲面表示, 具有更高的逼近精度和光滑性[34]. 这使得 IGA 在处理复杂数学模型和物理问题时具有明显优势. 相比之下, 传统 FEA 往往需要更高阶的单元或更细的网格来达到相同的精度. IGA 的高逼近精度有助于降低计算成本并提高计算效率.
3) 更好的处理复杂几何与拓扑结构: IGA 方法能够更好地处理复杂几何和拓扑结构, 如高曲率表面和多材料界面问题等. 在传统 FEA 中, 这些复杂结构需要经过繁琐的几何建模和网格生成过程, 容易出现网格质量问题. IGA 中精确和高阶光滑的曲线和曲面表示使得其在这些场景下具有明显优势, 如形状优化[27]、接触问题[35] 和多相流问题[36] 等.
4) 支持高阶连续性和高阶导数: IGA 能够很容易地实现高阶连续性和高阶导数的计算, 这在许多应用场景中具有重要意义, 如双调和方程[37,38] 和板壳分析问题[15,39] 等. 相比之下, 传统 FEA 要实现高阶连续性, 需要采用特殊的高阶单元, 这无疑会增加计算复杂度和实现难度. 而 IGA 天然地支持高阶连续性和高阶导数计算, 为解决复杂的物理问题提供了强大的工具.
5) 更强的局部精细化能力: 在传统 FEA 中, 局部细化可能会引入全局的网格扭曲和几何误差. 相比之下, IGA 具有更强的局部精细化能力, 可以在不影响全局几何精度的前提下, 对特定区域进行局部细化[40–45]. 这对于解决具有多尺度特征的

[1]Coreform IGA 是一款专门开发 IGA 的软件, 该软件支持线性和非线性问题的求解, 涵盖结构力学, 流体力学等多个领域, 软件主页: https://coreform.com.

[2]ANSYS LS-DYNA 是一款著名的商业有限元软件, 广泛应用于碰撞, 爆炸和金属成形等问题的求解. 近年来, LS-DYNA 开始引入 IGA 技术, 以支持更复杂的几何建模和分析. https://www.ansys.com/resource-center/webinar/ansys-2022-r1-ansys-ls-dyna-update.

[3]Rhino 是一款流行的三维建模软件, 而 Grasshopper 是其内置的参数化建模插件. 虽然它们本身不直接提供 IGA 功能, 但有第三方插件可实现 IGA, 如 Kiwi!3D. https://www.kiwi3d.com.

工程结构和局部应力集中等问题具有重要意义[46,47].

在现代 CAD 系统中, CAD 模型通常采用边界表示法 (Boundary Representation, 简称 B-Rep) 进行描述. B-Rep 通过顶点、边和面来定义物体的边界, 同时利用拓扑连接关系来组织这些基本元素. 这种表示法在几何建模和设计中具有很高的灵活性和效率. 图 1.3 左侧展示了一段管道的边界表示模型. 然而, 在进行等几何仿真分析之前, 我们首先需要构建模型内部的体表示法 (Volumetric Representation, 简称 V-Rep). 这一过程被称作区域参数化 (Domain Parameterization) 或计算域参数化 (Computational Domain Parameterization). 在本论文中, 简称为参数化. 在构造出适合分析的参数化结果之后, 便可以在体表示模型上进行等几何分析, 如图 1.3 中的右下图所示.

本论文将针对等几何分析中的区域参数化问题展开讨论. 首先, 我们将介绍适用于大多数问题的各向同性参数化方法, 该方法在保证单值性的前提下, 要求参数化结果具有尽可能良好的正交性和均匀性. 随后, 我们将探讨各向异性参数化方法. 这类方法允许在特定方向上调整参数化单元的密度和形状, 从而提高计算精度和效率.

1.2 国内外相关研究工作进展

在本节中, 我们将回顾与本研究相关的工作. 为了保持结构的清晰与简洁, 我们将本节划分为两部分, 分别介绍各向同性和各向异性参数化构造的相关研究. 图 1.4 展示了各向同性参数化和各向异性参数化之间的区别.

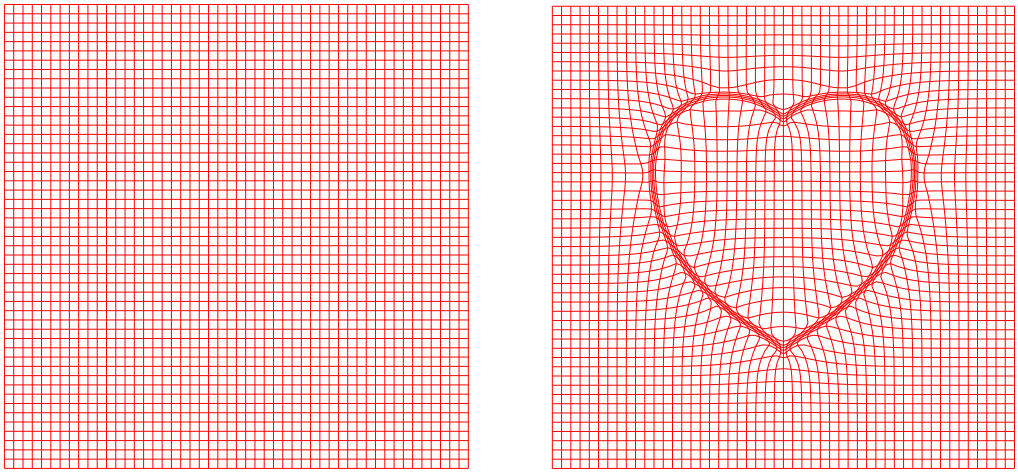


图 1.4 各向同性参数化 (左) 和各向异性参数化 (右)
Fig. 1.4 Isotropic Parameterization (Left) and Anisotropic Parameterization (Right)

1.2.1 适合分析的各向同性参数化方法

在进行等几何仿真分析之前, 关键的一步是由给定的 CAD 模型边界表示构造适合分析的样条参数化. 文献[48–50]指出, 参数化质量对后续分析的精度和效率具有重

要影响. 一个适合分析的参数化首先应当是一个双射 (或称满足单值性或有效性). 其次, 还应具备良好的正交性和均匀性. 或者等价地说, 具有尽可能小的角度和面积/体积畸变. 下面, 我们将现有的各向同性参数化方法按照处理双射约束的方式进行分类.

1) *代数参数化方法*. 这类方法最多求解一个线性方程组, 因此在计算上非常高效. 一个典型的代表是离散 Coons 方法[51], 它属于一种特殊的超限插值 (TransFinite Interpolation, TFI) 方法. 该方法是一种显式的参数化方法, 无需求解任何方程组, 通常被认为具有最高的计算效率. 徐岗等人将该方法推广到三维 NURBS 体参数化生成中[49]. 文献[52]讨论了几种线性参数化方法, 如弹簧模型方法, 均值坐标方法等, 这些方法需要求解一个线性方程组. 由于代数参数化方法没有专门考虑单值性约束, 在复杂计算域的参数化构造中经常产生具有自相交的结果. 因此, 在处理复杂区域的参数化问题时, 它们往往被用于计算一个初始猜测.

2) *非线性约束优化方法*. 这类方法的思路非常自然, 将参数化的单值性条件作为约束, 并以刻画参数化正交性和均匀性的能量函数作为目标泛函. 通过求解约束优化问题, 构造适合分析的参数化. 徐岗等人基于参数化的雅可比行列式可以用更高阶 NURBS 函数表示这一观察, 将雅可比行列式的非线性系数作为不等式约束以保证参数化结果的双射性[53]. 随后, 他们将此方法推广到了体参数化情形[49]. 利用约束聚合、分而治之和分层优化策略, 文献[54]提出了一个加速约束优化框架. 对于拓扑结构一致的计算域, 文献[55]引入了一种计算重用方法. 针对二次 B 样条参数化, Ugalde 等人提出了单值性的一系列充分条件和必要条件[56]. 尽管这些基于非线性约束优化的参数化方法在小规模算例上表现良好, 但当问题规模增大, 特别是在体参数化情况下, 其约束条件的数量相当庞大. 为了减轻计算负担, 潘茂东等人提出了一种基于配点的由粗到细添加约束的策略[2]. 然而, 该方法的计算量仍然十分惊人, 不得不借助商用优化求解器来求解这一问题.

3) *非线性无约束优化方法*. 在上述基于非线性约束优化的参数化方法中, 由于非线性约束的数量过于庞大, 造成求解上的困难. 因此, 近年来基于无约束优化的参数化方法变得流行起来. 徐岗等人[57] 提出了一种极小化变分调和映射的参数化方法. Nguyen 和 Jüttler 首先计算一个从计算域到参数域到调和映射序列, 随后采用样条逼近其逆映射[58]. 随后, Falini 等人[59] 将这一方法推广至二维 THB 样条参数化中, 首先利用边界元方法计算从计算域到参数域的调和映射, 然后采用样条最小二乘方法拟合其逆映射. 对于平面参数化情形, 年先顺与陈发来提出了一种基于 Teichmüller 映射的参数化方法. 潘茂东等人[1] 提出了一种基于拟共形映射的低秩参数化方法, 并采用交替方向乘子法极小化目标泛函, 随后将其推广至体参数化情形[60]. 最近, 张珊珊等人[61] 将这一方法应用于平面三角 Bézier 参数化中. Winslow 泛函极小值点的单值性由 Radó-Kneser-Choquet 定理保证, 因此它

在传统的网格生成领域广为应用. 在计算机图形学领域中常常被称为最等距参数化 (Most Isometric ParametrizationS, MIPS) 能量[62], 还有其三维版本[63] 也被计算机图形学领域所熟知. 这些能量泛函往往需要一个已经是双射的初始参数化. 为此, 研究人员提出了不同的消除折叠方法. 宿建平等人[64] 将雅可比矩阵投影到一个 K-扭曲有界空间, 随后刘浩等人[3] 进一步发展了该方法, 并考虑同时优化边界对应. 最近, 郑野和陈发来将这一思想应用到体 THB 样条参数化中, 提出了一种高效的体参数化方法[65]. 然而, 上述方法仍显复杂. 在本论文的第 3 章中, 我们提出了一种简单的消除折叠方法, 并将其应用于 NURBS 参数化中. 另一种思路是使用惩罚函数和雅可比正则化技术, 这可以追溯到文献[66,67]中的网格解扭问题. 王旭和马维银将这一思想应用到平面参数化问题当中, 有效地避免了额外的消除折叠步[68]. 在本论文的第 4 章中, 我们提出了一个新的惩罚函数以消除原惩罚函数的误差, 并且将其应用到构造适合分析的体参数化中.

4) *非线性偏微分方程方法*. 这类方法的基本思路是逼近满足已知边界条件的 Dirichlet 能量的驻点或者求解其对应的 Euler–Lagrange 方程. Martin 等人[69] 采用离散体调和映射来拟合三元 B 样条体. Shamanskiy 等人[70] 通过求解带有 neo-Hookean 超弹性材料定律的非线性弹性方程来构造适合分析的参数化. Ali 和马维银利用等几何配点方法求解带有边界导矢约束的偏微分方程构造平面参数化[71]. 基于椭圆网格生成 (Eliptic Grid Generation, EGG) 原理, Hinz 等人对 Laplace 方程进行离散, 提出了一系列基于非线性偏微分方程的参数化构造方法[72–74]. 这类基于 EGG 的椭圆参数化方法具有良好的收敛性质, 特别是在具有极端纵横比的狭长型计算域上表现良好. 然而, 这种方法存在两个主要问题. 第一, 对于一般的计算域, 它通常在边界曲率较大的凹/凸部分附近产生不均匀的单元. 第二, 文献[74]中采用标准 Newton 方法求解涉及到的非线性方程组. 该方法需要在每一步迭代中计算非线性方程组的雅可比行列式, 对大规模问题来说计算效率较低. 为解决这两个关键问题, 第 5 章给出拟调和映射在 Sobolev 空间 H^1 中的离散化, 并提出预条件 Anderson 加速算法来改善收敛性质并提高计算效率.

对于几何形状高度复杂或者边界上具有超过四个尖点或线 (C^0 连续) 的计算域, 尤其是具有高亏格的计算域, 往往需要多块曲面曲体来构造高质量参数化. 例如对于平面情形, 骨架提取方法[75,76]、逼近凸分解方法[77]、Polysquare 技术[78,79] 已经被提出; 对于体参数化情形, 体模版方法[49,80]、Pillow 操作[81]、中点细分方法[82]、简化的 l_1 Polycube 方法[83,84]. 对于高亏格情形, 如何计算分块至关重要, 这一问题属于计算拓扑范畴. 作者对这个问题的理解和研究尚浅, 本文不打算对这一问题进行过多探讨.

此外, 还有一些工作考虑非标准 NURBS 参数化技术, 如 T 样条[85,86]、(T)HB 样条[59,65,87]、PHT 样条[88]、toric 方法[89–91]、TCB 样条[92,93] 和细分曲面曲体方法[94,95].

1.2.2 适合分析的各向异性参数化方法

许多物理问题天然地具有局部特征和多尺度特性. 对于这类问题, 采用前述各向同性参数化方法往往效率较低. 为此, 许多研究者关注自适应的各向异性参数化方法, 希望能在这类问题中显著降低运行时间并节省内存开销. 这些方法主要分为两大类:

第一类是自适应加细方法. 这类方法试图在数值误差较大或数值解梯度较大的区域增加自由度. 传统的张量积型 B 样条和 NURBS 几何不具备局部细化能力, 每次加细都会引入大量冗余的自由度. 因此, 在这类方法中, 具有局部细化能力的样条工具变得至关重要. 例如, 层次 B 样条 (HB 样条)[40,96], 截断层次 B 样条 (THB 样条)[41,97,98], T 样条[42,85,86] 和截断 T 样条[99]. 还有基于升阶和降阶的 p 自适应方法, 以及基于基函数粗化或平滑的 k 自适应方法[11,14], 如具有局部 p 细化能力的混合次数加权 T 样条[100].

第二类方法是 r 自适应方法. 这类方法旨在通过重新分布内部控制顶点或调整内部权因子来改变参数化单元的位置和大小, 从而在不增加自由度的前提下降低计算误差. 据作者所知, 在等几何分析背景下, 这类方法最早可追溯至徐岗等人提出的 r 细化方法[53]. 尽管称之为细化方法, 但其保持自由度不变. 此外, 在经典有限元移动网格方法中, 采用监视函数来指定网格单元的大小和形状. 有兴趣的读者可以参考[101,102]及其引用文献. 徐岗等人通过调整内部控制点, 构建了一种以后验误差估计为目标函数的各向异性参数化方法[53,103]. Gravesen 等人提出了一个面向分析的参数化方法框架, 利用不同的误差估计可以得到不同的各向异性参数化方法[52]. 然而, 由于采用了单值性条件系数作为约束, 这一类方法仅适用于小规模问题. 随后, 徐岗等人通过结合 Winslow 映射, 绝对曲率和后验误差估计, 提出了一种新的 r 自适应方法[104]. 这种方法的单值性由 Winslow 项保证, 但在一定程度上牺牲了参数化质量. Ali 和马维银提出了一种依赖于解的等几何配点 PDE 方法用于构造各向异性参数化[105]. 对于多边形计算域 (边数 ≤4), 纪野等人利用曲率度量给出了一种保单值的自适应 toric 参数化方法, 其中计算域内部的权因子被用作优化变量[106]. 在本文的第 7 章中, 我们将提出一种采用绝对主曲率度量和双层优化策略的各向异性 NURBS 参数化新方法.

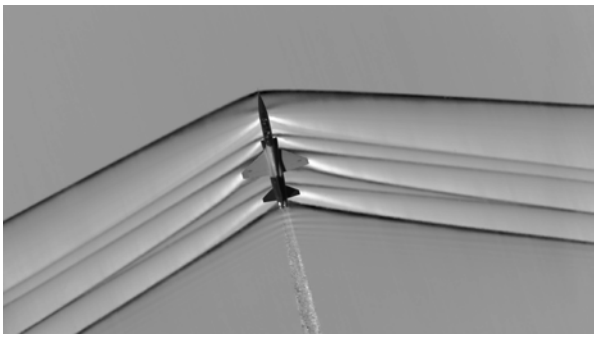
1.3 本文主要内容与贡献

大量研究表明[48–50]: 参数化质量对后续仿真分析的精度和效率具有显著影响. 从复杂计算域的边界表示出发, 构造适合分析的样条参数化是等几何分析中的公开问题[11,107]. 解决这一问题需要应对诸多挑战, 如:

1) 避免自相交: 在参数化过程中, 需要确保生成的参数化不会产生自相交, 即计算域内任意点的雅可比行列式不为零. 自相交的参数化可能会导致数值分析结果不准确或数值溢出从而计算终止, 并最终影响工程决策;
2) 提高参数化质量: 为了获得高精度的数值结果, 有必要提高参数化质量, 包括等

参数曲线的正交性、均匀性和光滑性等. 由于几何结构的复杂性, 选择合适的优化方法和策略是一个难题;

3) 提高计算效率: 在保证准确性的同时, 参数化问题求解需要保证计算效率. 针对大规模问题和实际工程应用, 需要开发高效的算法和技术以降低计算成本和时间消耗, 这对工程应用中结构的快速响应分析具有重要意义.



(a) 超音速激波[1]

(b) 燃烧[2]

(c) 海浪[3]

(d) 黄河三角洲[4]

图 1.5　自然界中广泛存在局部和多尺度特征

Fig. 1.5　Widespread Presence of Local and Multi-Scale Features in Nature

针对上述挑战, 本文研究了两类高效且高质量的参数化构造方法: 基于优化和基于非线性偏微分方程 (PDE) 的参数化方法. 在基于优化的参数化方法中, 我们首先考虑了一种障碍函数方法. 该方法从一个可能具有自交的初始参数化出发, 首先通过消除折叠获得一个近似单值的参数化, 然后通过最小化修正的加权目标泛函进一步提高

[1]图源: https://www.nasa.gov/sites/default/files/thumbnails/image/airbos_f7_p5.jpg.

[2]图源: https://en.wikipedia.org/wiki/Combustion#/media/File:Fire,_Fireplace,_Rostov-on-Don,_Russia.jpg.

[3]图源: https://en.wikipedia.org/wiki/Wind_wave#/media/File:Porto_Covo_Outubro_2014-3.jpg.

[4]图源: https://en.wikipedia.org/wiki/Yellow_River#/media/File:Yellow_River_Delta.jpeg.

参数化质量. 然而, 这种方法在消除折叠的第二步中, 不能充分利用目标泛函和梯度, 从而造成计算资源的浪费. 为解决这一问题, 我们提出了一种基于惩罚函数的参数化方法. 通过应用惩罚函数和雅可比正则化思想, 该方法能够在消除折叠的同时提升参数化质量, 避免了计算资源的浪费. 在后续研究中, 我们发现前述两种方法在具有极端纵横比的狭长计算域上表现不佳. 针对这一挑战, 我们提出了一种基于非线性 PDE 求解的椭圆参数化方法. 此外, 为了进一步提高求解效率, 我们还提出了一种预条件 Anderson 加速方法. 这种方法结合了预条件技术和 Anderson 加速策略, 有效地提高了参数化方法在处理具有极端纵横比的狭长计算域时的性能.

在前面关于各向同性参数化构造的讨论中, 我们希望参数化结果具有良好的正交性和均匀性, 如图 1.4 (左) 所示. 然而, 自然界中许多物理现象具有天然的局部和多尺度特性. 为了实现高保真模拟, 准确捕捉这些跨尺度的关键特征显得至关重要. 例如, 如图 1.5 所示, 在超音速飞行器中的激波现象, 燃烧中的火焰面, 海浪和黄河三角洲的两相界面等场景中都存在重要的局部特征. 对于这些情况, 采用前述的各向同性参数化进行计算往往效率较低. 因此, 在保证计算精度和计算量可接受的前提下, 对计算域进行自适应表示是非常有必要的. 图 1.4 (右) 展示了一个各向异性的参数化结果.

近年来, 许多研究开始关注自适应等几何分析方法[108]. 在这方面, 大部分研究者主要讨论等几何分析中的后验误差估计, 然后利用具有局部加细能力的样条工具 (如 THB 样条[40,97,98], T 样条[109], PHT 样条[110] 等) 进行局部细化. 这些方法通常在某些关注的区域、解后验误差估计或梯度较大的区域进行局部加细. 与这些方法不同, 本研究考虑了一种 r 自适应方法, 其特点在于沿不同方向具有不同的缩放或变形特性. 其目标为在不增加自由度的前提下, 通过重新分布内部控制顶点的位置来降低等几何分析数值解的误差. 这样一来, 既提高了等几何数值解的精度, 又保持了原有的自由度, 有助于降低计算成本. 有研究表明[111], 在计算流体力学仿真分析中采用与流场方向一致的结构网格有助于避免数值扩散. 此外, 将等几何解曲面的曲率信息作为误差估计子, 并采用双层优化策略以进一步提高计算效率.

1.4 本文结构安排

本文后续各章内容安排如图 1.6 所示, 详细安排如下:

第 2 章简要介绍本文相关的预备知识, 旨在统一文中所涉及到的符号和术语. 其中包括 NURBS 曲线曲面、等几何分析基础以及适合分析的参数化问题描述.

第 3 章介绍一种基于控制顶点和权因子交替优化的障碍函数参数化方法. 该方法采用三阶段策略: 首先通过高效的代数参数化方法构造初始参数化; 然后计算一个在所有 Gauss 积分点上雅可比行列式为正的近似单值参数化; 最后通过极小化修正的加权泛函进一步提升参数化质量.

在第 3 章所提出的障碍函数方法中, 其第二步仅针对具有自相交的区域进行调整, 而对整体参数化质量提升有限, 造成了一定的计算浪费. 因此, 在第 4 章中, 基于惩罚函数和雅可比正则化技术提出了一种构造高质量参数化的高效方法. 该方法在消除自相交的同时, 能够有效提高参数化质量, 从而减少障碍函数方法中的计算资源浪费.

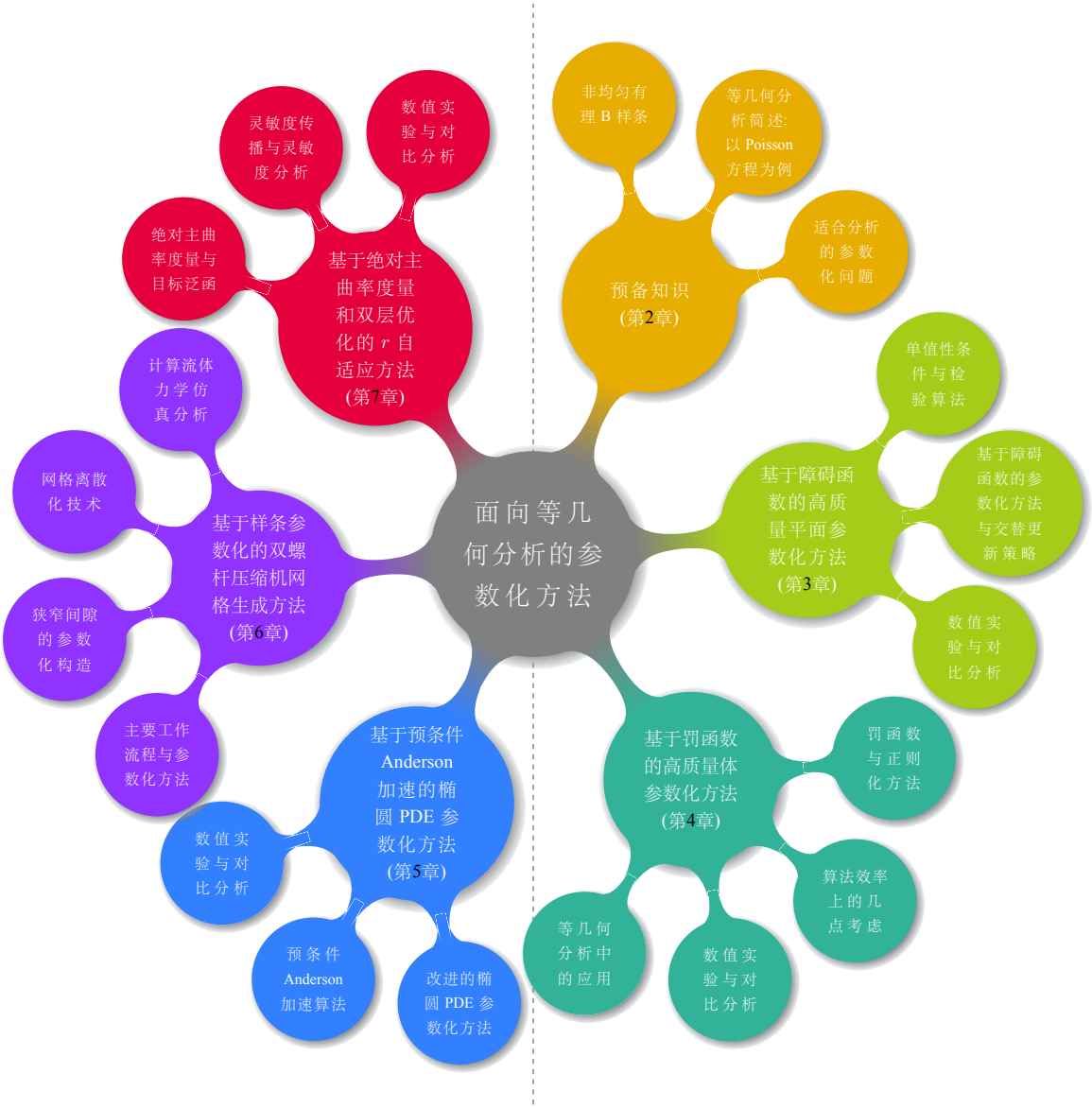


图 1.6　本文主要内容和结构安排
Fig. 1.6　Main Content and Structural Organization

在后续研究中, 我们发现前述两种基于优化的方法 (第 3 章的障碍函数方法和第 4 章的惩罚函数方法) 在具有极端纵横比的狭长计算域上表现不佳. 为此, 在第 5 章中, 我们考虑一种改进的基于椭圆 PDE 的参数化方法. 该方法能为具有狭长特征的计算域构造满意的参数化结果. 为加速求解非线性方程组, 我们提出一种预条件 Anderson 加速方法. 此外, 在这一章中, 我们对比了前述三种各向同性参数化方法.

在第 6 章中, 我们将第 5 章中所提出的改进椭圆参数化方法应用到真实工业场景中双螺杆压缩机的网格生成问题中. 首先为该类型机械生成高阶光滑的样条参数化, 随后可以很容易地将所构造的样条体参数化转化为具有不同性质的结构六面体网格, 如边界层网格和流对齐网格等. 在商用软件 ANSYS CFX ™ 中的仿真结果表明, 本文方法生成的结构六面体网格可以适用于复杂的计算流体力学仿真分析.

前述各向同性参数化对一些具有局部特征的物理问题中表现不佳. 为此, 考虑在不增加自由度的前提下, 达到尽可能高的精度, 以期望有效降低计算成本. 在第 7 章中, 我们考虑一种基于曲率度量的各向异性参数化方法. 为平衡该方法的精度和计算效率, 提出了一种双层优化策略, 即在较粗的设计模型上更新参数化, 而在较细的模型上进行分析. 数值实例证明了该方法的有效性以及其在各向异性问题中的优势.

第 8 章将总结本论文的工作, 并展望未来的研究方向, 讨论潜在的研究课题和改进空间. 希望能够为后续研究提供启示和方向, 进一步推动本领域的发展.

2 预备知识

本章简要介绍相关的预备知识, 主要目的是统一文中所涉及的符号和术语.

2.1 非均匀有理 B 样条

非均匀有理 B 样条 (NURBS) 是具有一定光滑性的分段有理多项式函数, 它在 CAD 领域应用十分广泛. 在本小节中, 我们将简要回顾 NURBS 的定义和基本性质.

首先介绍节点向量的概念. 节点向量 Ξ 是一个非递减的实数序列

$$\Xi = \{\xi_0, \xi_1, \ldots, \xi_{n+p+1}\}, \quad \text{满足 } \xi_0 \le \xi_1 \le \cdots \le \xi_{n+p+1},$$

其中 $\xi_i \in [0,1]$ 是第 i 个节点, n 是基函数的个数, p 是基函数的次数. 我们称半开区间 $[\xi_i, \xi_{i+1})$ 为第 i 个节点区间. 注意这里某些节点 ξ_i 可能相等, 因此某些节点区间的长度可能为零. 为了便于区分, 一般将节点向量中不同节点值所构成的序列称为断点 (breakpoint). 在等几何分析中, 将具有非零长度的节点区间或者连续两个断点之间的区间称为单元. 如果一个节点 ξ_i 出现 m 次 (即 $\xi_i, \xi_{i+1}, \ldots, \xi_{i+m-1}$), 其中 $(m > 1)$, 我们说节点 ξ_i 的重数或重复度为 m. 此外, 如果节点向量 Ξ 的第一个和最后一个节点值的重数是 $p+1$, 那么我们称 Ξ 是开的 (open 或 clamped 或 nonperiodic).

一个 p 次 B 样条基函数 $N_i^p(\xi)$ 可以由 Cox-de Boor 公式递推定义[8]:

$$N_i^p(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_i^{p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1}^{p-1}(\xi), \quad i = 0, 1, \ldots, n, \tag{2.1}$$

其中零次 B 样条基函数定义为

$$N_i^0(\xi) = \begin{cases} 1, & \text{如果 } \xi \in [\xi_i, \xi_{i+1}), \\ 0, & \text{其它}. \end{cases}$$

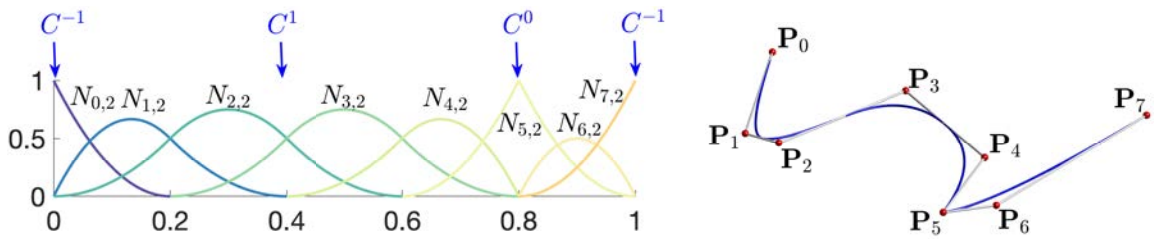
这里我们约定比值 $0/0$ 为零.

B 样条基函数具有许多良好的性质, 如非负性、单位分解性、紧支集和光滑性. 这里需要特别说明的是, 基函数的光滑性与节点值的重数有关. 一个 p 次的基函数在节点 ξ_i 处是 $p - m_i$ 阶光滑的, 这里 m_i 是该节点值的重数. 当 $m_i = p$ 时, 则基函数 $N_{i,p}$ 插值于这一节点. 图 2.1(a) 展示了定义在开节点区间 $\Xi^* = \{0, 0, 0, 0.2, 0.4, 0.6, 0.8, 0.8, 1, 1, 1\}$ 上的二次 B 样条基函数, 节点值 0.8 的重数为 2. 因此在 $\xi = 0.8$ 处, 基函数 $N_5^2(\xi)$ 是 C^0 连续的, 且值为 1, 而其余基函数值为 0.

形式上, 记由定义在节点向量 Ξ 上的 p 次 B 样条基函数所张成的样条空间为:

$$\mathbb{S}_p^{\Xi} = \left\{\psi \in L^2([\xi_p, \xi_{n+1}]) : \ \psi|_{\xi=\xi_i} \in C^{k_i}, \ \psi|_{\xi\in(\xi_i,\xi_{i+1})} \in \mathbb{P}^p\right\}, \tag{2.2}$$

其中, $\mathbb{P}^p$ 表示 p 次多项式的空间, $k_i = p - m_i$ 是光滑阶, m_i 表示第 i 个节点值的重数.



(a) B 样条基函数　　(b) 一条 B 样条曲线

图 2.1　开节点区间 $\Xi^* = \{0,0,0,0.2,0.4,0.6,0.8,0.8,1,1,1\}$ 上的二次 B 样条基函数和曲线
Fig. 2.1　Quadratic B-spline Basis Functions and Curve Defined over the Open Knot Vector $\Xi^* = \{0,0,0,0.2,0.4,0.6,0.8,0.8,1,1,1\}$

空间 $\mathbb{R}^d$ 中的一条 p 次 B 样条曲线是样条空间 $\mathbb{S}_p^{\Xi}$ 中的向量值函数:

$$\mathcal{C}(\xi) = \sum_{i=0}^{n} \mathbf{P}_i N_i^p(\xi), \ \xi \in \hat{\Omega} = [\xi_p, \xi_{n+1}], \tag{2.3}$$

其中 $\mathbf{P}_i \in \mathbb{R}^d$ 表示控制顶点, 我们称 $\hat{\Omega}$ 为该 B 样条曲线的参数域. 在实际应用中, 一般将参数区间取为 $[0,1]$. 在本文中, 如不加特殊说明, 总假定参数区间为 $[0,1]$.

图 2.1(b) 展示了一条定义在开节点向量 Ξ^* 上的二次 B 样条曲线. 该曲线插值于控制顶点 $\mathbf{P}_5$, 这是因为这一节点值的重数等于该曲线的次数 2. 同时, 该曲线插值于两端的控制顶点 $\mathbf{P}_0$ 和 $\mathbf{P}_7$. 这是定义在开节点向量上 NURBS 曲线的一个典型特征. 在后续的介绍中我们还将会看到, 使用开节点向量的另一个优势是: 一个定义在 d 维参数域上的 NURBS 对象的边界是由其边界控制顶点和基函数所定义的 $d-1$ 维 NURBS 对象. 这一特点极大地方便了设计人员对 CAD 产品的外观设计, 因此开的节点向量在现代 CAD 系统中广泛使用. 在本文中, 如果没有特殊说明, 总是假定节点向量是开的.

曲面是一个二维流形, 具有两个参数方向. 为了定义空间 $\mathbb{R}^d$ 中的 B 样条曲面, 首先在 ξ 和 η 参数方向上分别定义开的节点向量 Ξ 和 $\mathcal{H}$. 对每一个节点向量, 我们可以按照式 (2.1) 分别定义 p 次单变量 B 样条基函数 $\left\{N_{i_1}^p(\xi)\right\}_{i_1=0}^{n_1}$ 和 q 次单变量 B 样条基函数 $\left\{N_{i_2}^q(\xi)\right\}_{i_2=0}^{n_2}$. 于是, 我们可以定义双变量张量积型 B 样条基函数:

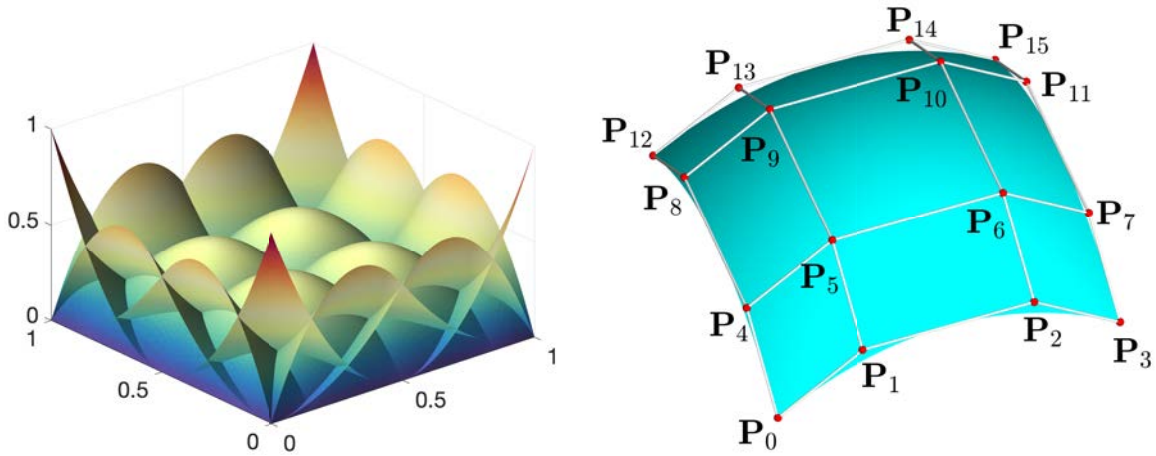
$$N_{i_1,i_2}^{p,q}(\xi,\eta) = N_{i_1}^p(\xi) N_{i_2}^q(\eta), \ i_1 = 0,1,\ldots,n_1, \ i_2 = 0,1,\ldots,n_2. \tag{2.4}$$

图 2.2(a) 展示了在节点向量 $\Xi = \{0,0,0,0.5,1,1,1\}$ 和 $\mathcal{H} = \{0,0,0,0.5,1,1,1\}$ 上定义的双二次张量积 B 样条基函数.

与 B 样条曲线定义类似, $p \times q$ 次张量积 B 样条曲面可以定义为双变量基函数和控制顶点的线性组合:

$$\mathcal{S}(\xi,\eta) = \sum_{i_1=0}^{n_1}\sum_{i_2=0}^{n_2} \mathbf{P}_{i_1,i_2} N_{i_1,i_2}^{p,q}(\xi,\eta),\ (\xi,\eta) \in \hat{\Omega} = [\xi_p, \xi_{n_1+1}] \times [\eta_q, \eta_{n_2+1}], \tag{2.5}$$

其中 $\hat{\Omega}$ 被称为 B 样条曲面的参数域. 如未特殊说明, 对于平面情形我们假定 $\hat{\Omega} = [0,1]^2$.



(a) 张量积 B 样条基函数　　(b) 张量积 B 样条曲面

图 2.2　双二次张量积 B 样条基函数和曲面

Fig. 2.2　Biquadratic Tensor-Product B-spline Basis Functions and Surface

为了叙述方便, 有时我们将上式 (2.5) 中的张量索引转换为全局线性索引, 即

$$\mathcal{S}(\xi,\eta) = \sum_{i=0}^{n} \mathbf{P}_i N_i(\xi,\eta), \tag{2.6}$$

其中 $n = (n_1+1)(n_2+1) - 1$. 图 2.2(b) 展示了一张 B 样条曲面. 注意, 这里两个参数方向上均采用开的节点向量, 其边界曲线的几何形状完全由该条边界上的基函数和控制顶点所决定. 例如, 该曲面的左侧边界曲线是由节点向量 $\mathcal{H}$ 和控制顶点 $\mathbf{P}_0$, $\mathbf{P}_4$, $\mathbf{P}_8$, 和 $\mathbf{P}_{12}$ 所确定的一条 B 样条曲线.

类似地, 给定三个参数方向上的节点向量 Ξ, $\mathcal{H}$ 和 $\mathcal{Z}$, 我们可以定义如下的 $p \times q \times r$ 次三变量张量积 B 样条体

$$\mathcal{V}(\xi,\eta,\zeta) = \sum_{i_1=0}^{n_1}\sum_{i_2=0}^{n_2}\sum_{i_3=0}^{n_3} \mathbf{P}_{i_1,i_2,i_3} N_{i_1,i_2,i_3}^{p,q,r}(\xi,\eta,\zeta),\ (\xi,\eta,\zeta) \in \hat{\Omega}, \tag{2.7}$$

其中 $\hat{\Omega} = [\xi_p, \xi_{n_1+1}] \times [\eta_q, \eta_{n_2+1}] \times [\zeta_r, \xi_{n_3+1}]$ 是参数域. 图 2.3 展示了一个张量积 B 样条体, 它在每个参数方向上的次数均是 2. 对于体情形, 本文假定 $\hat{\Omega} = [0,1]^3$.

图 2.3　一个 $2 \times 2 \times 2$ 次张量积 B 样条体
Fig. 2.3　A $2 \times 2 \times 2$ Tensor-Product B-spline Solid

尽管 B 样条几何在自由曲线曲面造型中具有较好的灵活性, 但是无法精确表示圆锥曲线曲面 (如椭圆, 双曲线). 而这些曲线曲面在工业场景中极其常见. 为解决这一问题, 我们需要将 B 样条推广至 NURBS. 一个 $\mathbb{R}^d$ 空间中的 NURBS 几何可由 $\mathbb{R}^{d+1}$ 中的 B 样条几何通过中心投影变换得到. 事实上, 圆锥曲线可以由分片二次多项式曲线经中心投影变换后精确表示, 因此可以由二次 NURBS 曲线精确表示.

对每个控制顶点 $\mathbf{P}_i \in \mathbb{R}^d$, 给它赋上一个权因子 $\omega_i \in \mathbb{R}^+$, 称

$$\mathbf{P}_i^\omega = \begin{pmatrix} \omega_i \mathbf{P}_i \\ \omega_i \end{pmatrix} \in \mathbb{R}^{d+1}$$

为该控制顶点的齐次坐标形式或射影控制顶点. 利用式 (2.3) 中的定义, 这组射影控制顶点定义了一条 $\mathbb{R}^{d+1}$ 中的 p 次 B 样条曲线 $\mathcal{C}^\omega(\xi)$, 称为射影曲线. 那么, p 次 NURBS 曲线可以由该射影曲线 $\mathcal{C}^\omega(\xi)$ 经过中心投影变换得到:

$$\mathcal{C}(\xi) = \sum_{i=0}^{n} \mathbf{P}_i R_i(\xi), \quad \xi \in \hat{\Omega}, \tag{2.8}$$

其中

$$R_i(\xi) = \frac{\omega_i N_i^p(\xi)}{\sum_{i=0}^{n} \omega_i N_i^p(\xi)} \tag{2.9}$$

被称为 NURBS 基函数. 图 2.4 展示了由 $\mathbb{R}^3$ 中的二次 B 样条曲线经中心投影变换得到

的 $\mathbb{R}^2$ 中圆的 NURBS 表示. NURBS 曲面和曲体可以通过类似的方式定义.

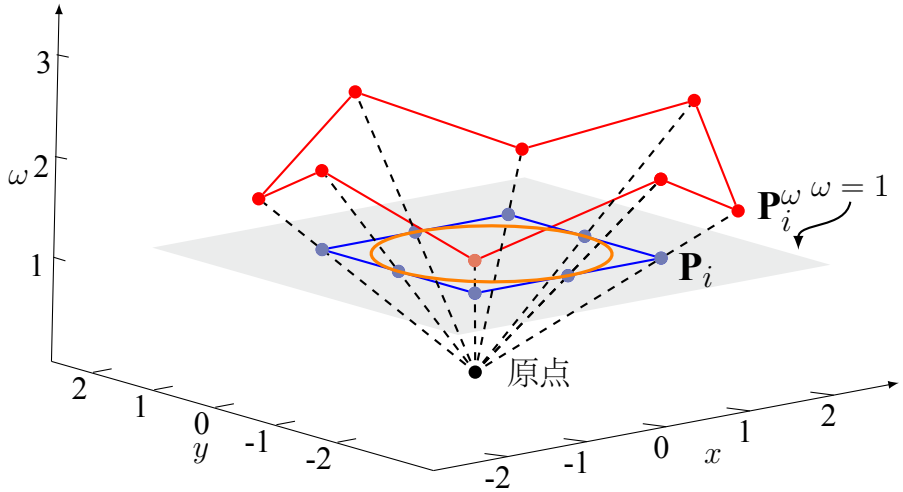


图 2.4　$\mathbb{R}^2$ 中圆的 NURBS 表示由 $\mathbb{R}^3$ 中的二次 B 样条曲线经中心投影变换得到

Fig. 2.4　NURBS-represented Circle in $\mathbb{R}^2$ via Central Projection of a Quadratic B-spline Curve in $\mathbb{R}^3$

根据式 (2.9) 中 NURBS 基函数的定义和 B 样条基函数的单位分解性质, 我们可以很容易地知道: 若所有的权因子 ω_i 均相同, 那么 NURBS 曲线退化为 B 样条曲线. 因此 B 样条是 NURBS 的特殊情形. 类似地, NURBS 曲面和曲体的定义可以很容易地得到.

在传统有限元方法中, 细化物理网格绝非易事. 而在等几何分析方法中, 借助于 NURBS 表示的节点插入和升阶算法, 可以很容易地在保证几何精确性的前提下实现细化操作[11]. 在等几何分析中, 有三种常见的细化方法:

- h-细化方法: 借助于 NURBS 表示的节点插入算法;
- p-细化方法: 借助于 NURBS 表示的升阶算法;
- k-细化方法: 先执行 NURBS 升阶算法, 再执行 NURBS 节点插入算法.

有关 NURBS 更加深入详尽的介绍, 感兴趣的读者请参阅文献[4,7,8].

2.2　等几何分析简述: 以 Poisson 方程为例

本节以 Poisson 方程为例, 简要介绍等几何分析的基本计算流程. 对于熟悉有限元分析的读者, 这部分很容易理解. 图 2.5 展示了等几何分析计算流程示意图.

作为模型问题, 考虑如下的 Poisson 方程:

$$\Delta \mathscr{U} + f = 0, \quad \boldsymbol{x} \in \Omega, \tag{2.10a}$$

$$\mathscr{U} = g, \quad \boldsymbol{x} \in \Gamma_D, \tag{2.10b}$$

$$\nabla \mathscr{U} \cdot \mathbf{n} = h, \quad \boldsymbol{x} \in \Gamma_N, \tag{2.10c}$$

其中 $\Omega \subset \mathbb{R}^d$ 是物理域, Γ_D 表示边界上满足 Dirichlet 边界条件的部分, Γ_N 表示边界上满足 Neumann 边界条件的部分, 且满足 $\overline{\Gamma_D \bigcup \Gamma_N} = \partial\Omega$, $\Gamma_D \bigcap \Gamma_N = \varnothing$, $\mathbf{n}$ 是边界上的单位外法线向量, $f = f(\boldsymbol{x}) \in L^2(\Omega) : \Omega \to \mathbb{R}$ 是给定的源项, $\mathscr{U}(\boldsymbol{x}) : \Omega \to \mathbb{R}$ 是待求的解.

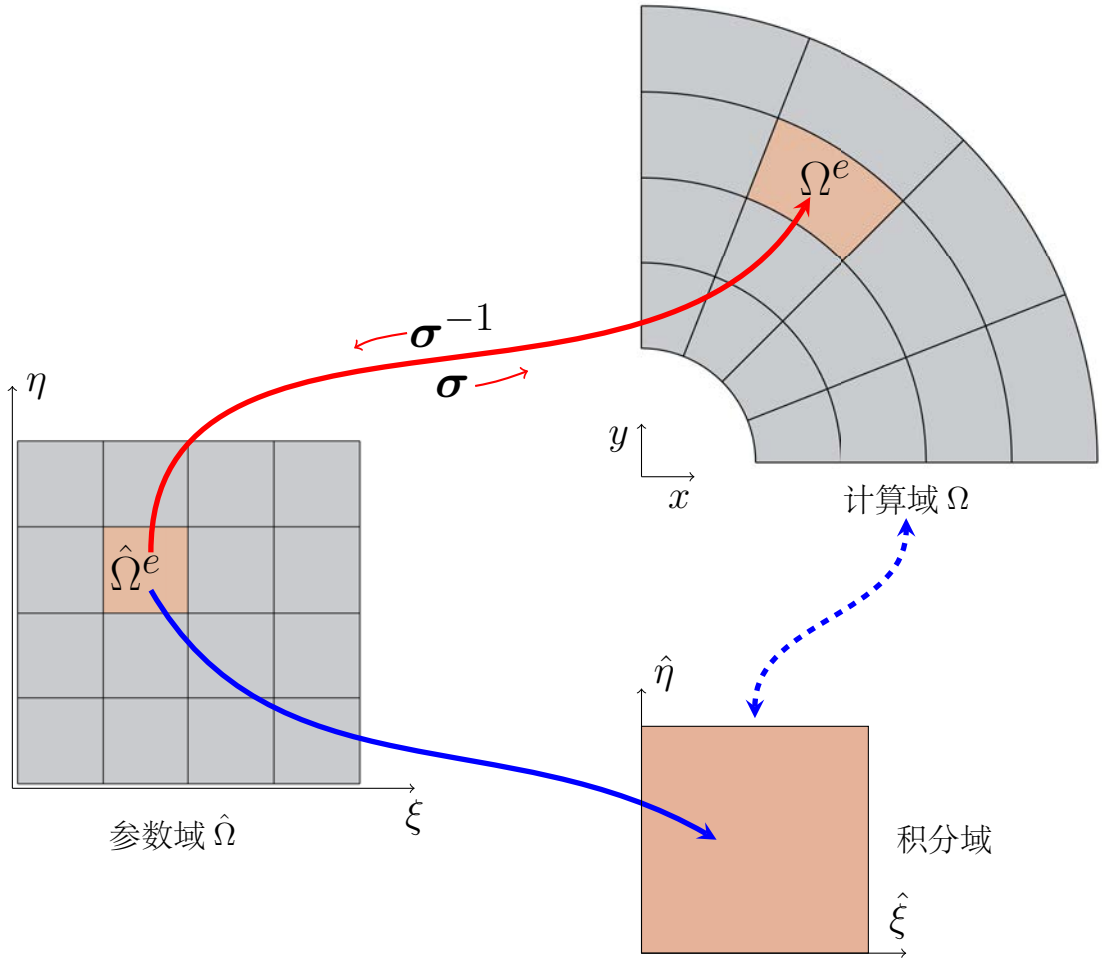


图 2.5　等几何分析计算示意图

Fig. 2.5　Schematic Diagram of Isogeometric Analysis

对于上述问题 (2.10), 其弱形式为: 寻找 $\mathscr{U} \in H^1(\Omega)$, 使得

$$a(\mathscr{U}, \mathscr{V}) = f(\mathscr{V}), \quad \forall\, \mathscr{V} \in H^1_{0,\Gamma_D}(\Omega) \tag{2.11}$$

成立, 其中

$$a(\mathscr{U}, \mathscr{V}) = \int_\Omega \nabla\mathscr{U} \cdot \nabla\mathscr{V} \mathrm{d}\Omega, \tag{2.12}$$

$$f(\mathscr{V}) = \int_\Omega f\mathscr{V} \mathrm{d}\Omega + \int_{\Gamma_N} h\mathscr{V} \mathrm{d}s, \tag{2.13}$$

$H^1_{0,\Gamma_D}(\Omega) \subset H^1(\Omega)$ 是 Sobolev 空间 $H^1(\Omega)$ 中满足在边界 Γ_D 上为零的子函数空间.

依照 Garlekin 方法, 我们可以将上述弱形式 (2.11) 离散为: 寻找 $\mathscr{U}^{\mathrm{h}} \in \mathbb{V}^{\mathrm{h}}$, 使得

$$a(\mathscr{U}^{\mathrm{h}}, \mathscr{V}^{\mathrm{h}}) = f(\mathscr{V}^{\mathrm{h}}), \quad \forall\, \mathscr{V}^{\mathrm{h}} \in \mathbb{V}^{\mathrm{h}}_{0,\Gamma_D}(\Omega) \tag{2.14}$$

成立, 其中 $\mathbb{V}^{\mathrm{h}}$ 是 $H^1(\Omega)$ 的有限维子空间, $\mathbb{V}^{\mathrm{h}}_{0,\Gamma_D}(\Omega)$ 是 $H^1_{0,\Gamma_D}(\Omega)$ 的有限维子空间.

在等几何分析中, 将试探函数 $\mathscr{U}^{\mathrm{h}}$ 取为

$$\mathscr{U}^{\mathrm{h}} = \mathscr{U}^{\mathrm{h}}(\boldsymbol{x}) = \sum_{i=0}^{n} \mathscr{U}_i R_i \circ \sigma^{-1}(\boldsymbol{x}), \tag{2.15}$$

这里 $\mathscr{U}_i$ 是未知的系数, 未知系数的个数 $n+1$ 一般被称为自由度 (Degrees-of-Freedom, 简称 DOFs), σ 是从参数域 $\hat{\Omega}$ 到计算域 Ω 的映射, σ^{-1} 表示其逆映射, 见图 2.5.

现在, 将测试函数 $\mathscr{V}^{\mathrm{h}}$ 取为 $R_i \circ \sigma^{-1}(\boldsymbol{x})$, 并将式 (2.15) 代入离散弱形式 (2.14) 中, 我们得到如下的线性方程组

$$\mathbf{K}\mathscr{U} = \mathbf{f}, \tag{2.16}$$

其中 $\mathscr{U} = [\mathscr{U}_0, \mathscr{U}_1, \ldots, \mathscr{U}_n]$ 是待求的向量, 刚度矩阵 $\mathbf{K} = \mathtt{A}\ \mathbf{K}^e$, 其中 $\mathtt{A}$ 表示矩阵组装算子, $\mathbf{K}^e$ 表示单元刚度矩阵, 其元素为

$$\begin{aligned} \mathbf{K}^e_{i,j} &= \int_{\Omega} \nabla R_i(\boldsymbol{x}) \cdot \nabla R_j(\boldsymbol{x})\, \mathrm{d}\Omega \\ &= \int_{\hat{\Omega}^e} \left(\boldsymbol{\mathcal{J}}^{-\mathrm{T}}(\boldsymbol{\xi}) \nabla_{\boldsymbol{\xi}} R_i(\boldsymbol{\xi})\right) \cdot \left(\boldsymbol{\mathcal{J}}^{-\mathrm{T}}(\boldsymbol{\xi}) \nabla_{\boldsymbol{\xi}} R_j(\boldsymbol{\xi})\right) |\boldsymbol{\mathcal{J}}(\boldsymbol{\xi})|\ \mathrm{d}\hat{\Omega}^e, \end{aligned} \tag{2.17}$$

右端向量 $\mathbf{f} = \mathtt{A}\ \mathbf{f}^e$, 其分量为

$$\begin{aligned} \mathbf{f}^e_i &= \int_{\Omega} f(\boldsymbol{x}) R_i \circ \sigma^{-1}(\boldsymbol{x})\, \mathrm{d}\Omega \\ &= \int_{\hat{\Omega}^e} f(\boldsymbol{x}(\boldsymbol{\xi})) R_i(\boldsymbol{\xi})\, |\boldsymbol{\mathcal{J}}(\boldsymbol{\xi})|\ \mathrm{d}\hat{\Omega}^e, \end{aligned} \tag{2.18}$$

这里 $\boldsymbol{\mathcal{J}}$ 表示从参数域 $\hat{\Omega}$ 到计算域 Ω 的映射 σ 的雅可比矩阵, $|\boldsymbol{\mathcal{J}}(\boldsymbol{\xi})|$ 表示其行列式.

求解线性方程组 (2.16) 可得未知向量 $\mathscr{U}$, 从而根据式 (2.15) 得到数值解 $\mathscr{U}^{\mathrm{h}}$.

2.3 适合分析的参数化

参数化是指从参数域 $\hat{\Omega}$ 到计算域 Ω 的映射 σ, 如图 2.5 所示. 在等几何分析中, 我们要求映射 σ 也由样条表示. 由式 (2.15) 和 (2.16) 可知, 分析所需的参数化要求映射 σ 是一个双射, 即存在逆映射. 事实上, 由于计算域边界表示是给定的且映射是连续的, 参数化的满射性质比较容易满足, 为了强调这一点有时也称为单值性. 在本论文剩余

部分中, 为了表述和理解上的便利, 在不产生混淆的前提下, 我们不再区分参数化和计算域 Ω 上的坐标, 而采用符号 $\boldsymbol{x}$ 来统一表示参数化映射本身和计算域上的空间坐标.

2.3.1 适合分析参数化问题的提出

为了表述上的清晰, 我们将样条参数化 $\boldsymbol{x}(\boldsymbol{\xi})$ 重写成如下形式:

$$\boldsymbol{x}(\boldsymbol{\xi}) = \underbrace{\sum_{i\in\mathcal{I}_I} \mathbf{P}_i R_i(\boldsymbol{\xi})}_{\text{未知}} + \underbrace{\sum_{j\in\mathcal{I}_B} \mathbf{P}_j R_j(\boldsymbol{\xi})}_{\text{已知}}. \tag{2.19}$$

即我们把 $\boldsymbol{x}(\boldsymbol{\xi})$ 分成两部分, 其中 $\mathbf{P}_i, i \in \mathcal{I}_I$ 是未知的内部控制顶点, $\mathbf{P}_j, j \in \mathcal{I}_B$ 是已知的边界控制顶点, $\mathcal{I}_I$ 和 $\mathcal{I}_B$ 分别是内部和边界控制顶点的索引集合. 类似地, $R_i(\boldsymbol{\xi}), i \in \mathcal{I}_I$ 和 $R_j(\boldsymbol{\xi}), j \in \mathcal{I}_B$ 分别是内部控制顶点和边界控制顶点所对应的基函数. 图 2.6 展示了构造适合分析参数化问题的示意图.

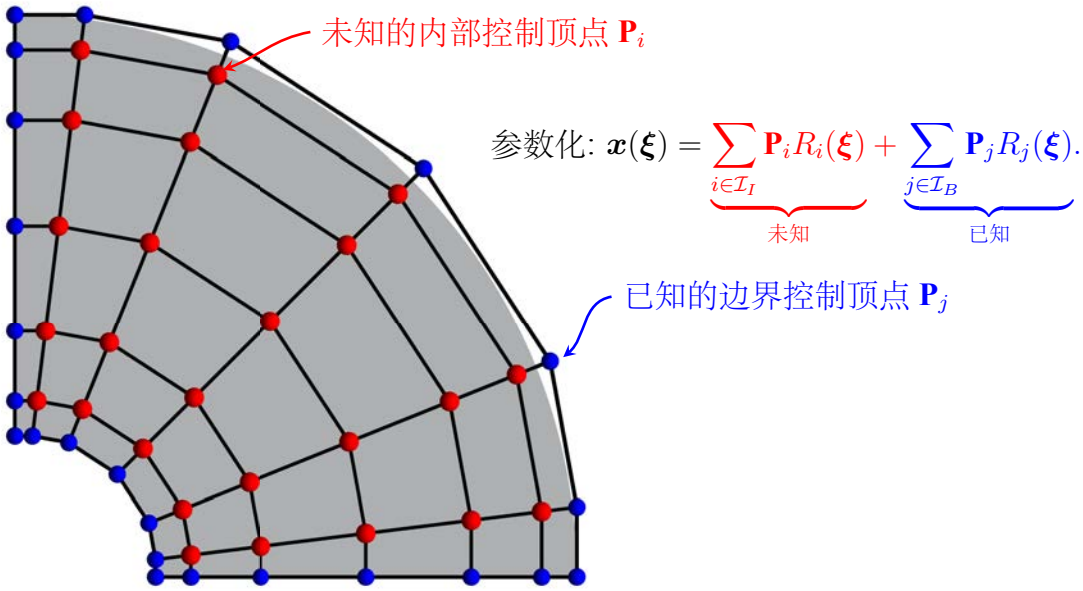


图 2.6 适合分析参数化问题的示意图
Fig. 2.6 Schematic Diagram of Analysis-Suitable Parameterization Problem

因此, 面向等几何分析的适合分析参数化问题可以归结如下: 由给定的边界控制顶点 $\mathbf{P}_j, j \in \mathcal{I}_B$ 出发, 构造内部的控制顶点 $\mathbf{P}_i, i \in \mathcal{I}_I$ 或基函数 $R_i(\boldsymbol{\xi}), i \in \mathcal{I}_I$, 使得参数化 $\boldsymbol{x}(\boldsymbol{\xi})$ 是一个双射, 且具有尽可能良好的正交性和均匀性.

下面的引理说明了一般参数化的单值性条件:

引理 2.1. ([112]) 假设 $\boldsymbol{x}$ 是参数域 $\hat{\Omega}$ 到计算域 Ω 的 C^1 参数化, 且它在计算域边界上的限制 $\boldsymbol{x}|_{\partial\hat{\Omega}}$ 是单值的. 那么如果其雅可比矩阵 $\boldsymbol{\mathcal{J}}$ 在参数域 $\hat{\Omega}$ 内部的所有点均可逆, 则

该参数化 $\boldsymbol{x}$ 是单值的. 换言之, 如果参数化 $\boldsymbol{x}$ 的雅可比行列式 $|\mathcal{J}|$ 在整个参数域内部均不为零, 那么该参数化 $\boldsymbol{x}$ 是一个单射.

由参数化 $\boldsymbol{x}$ 的连续性, 上述引理2.1要求参数化 $\boldsymbol{x}$ 的雅可比行列式在整个参数域 $\hat{\Omega}$ 上具有相同的符号. 否则, 若其雅可比行列式在参数域上有正有负, 那么由连续函数的介值定理, 可以立即推出其雅可比行列式在参数域内部至少一点等于零. 因此, 上述引理要求参数化 $\boldsymbol{x}$ 的雅可比行列式在整个参数域 $\hat{\Omega}$ 上具有相同的符号. 对于单值的参数化, 通过恰当地选取参数方向, 我们总可以将其雅可比行列式 $|\mathcal{J}|$ 在参数域 $\hat{\Omega}$ 的所有点上均大于零. 因此, 本论文仅就 $|\mathcal{J}| > 0$ 的情况进行讨论.

2.3.2 适合分析参数化存在的前提条件

从给定的边界表示出发, 构造高质量适合分析的参数化绝非易事. 特别是, 这一问题的解未必总是存在的. 为了保证解的存在性, 有必要对给定的边界表示做一些假设:

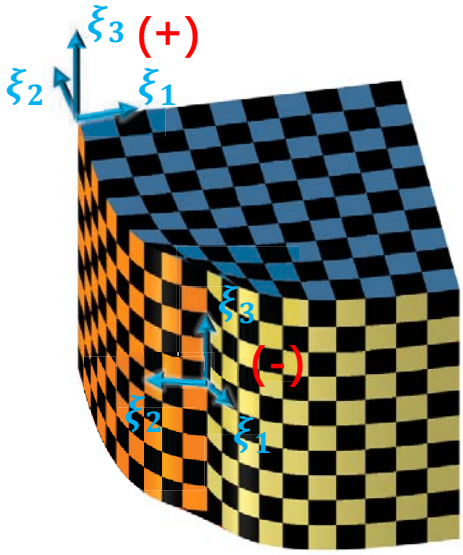


图 2.7 不可能由单块 C^k $(k \geq 1)$ 单值 NURBS 体参数化的计算域
Fig. 2.7 A Domain Not Representable by a Single C^k $(k \geq 1)$ Bijective NURBS Solid

- 对齐条件: 由于 NURBS 曲面和曲体具有张量积结构的限制, 相对的边界曲线或曲面必须具有相同的次数和节点向量. 若不满足此条件, 则需要进行一些预处理操作, 例如节点插入或升阶, 以实现对齐.
- 边界正则性条件[80]: 在二维情况下, 要求给定的边界曲线不能有自相交, 并且各曲线仅在端点处相交. 在三维情况下, 要求边界曲线或曲面不能有自相交, 并且各曲面只能在边界曲线上相交.
- 双射参数化的存在性: 虽然上述的边界正则性条件是必要的, 但它们并不足以保证计算域具有单值且 C^k $(k \geq 1)$ 连续的参数化. 在二维情况下, 文献[52]举出了

一个具有尖锐凹角的反例. 三维情形则更为复杂, 除了需要考虑 8 个角点处的正则性外, 还需考虑 12 条边界边处的正则性[113]. 如图 2.7 所示, 这一计算域沿两个给定的边界边的雅可比行列式具有相反的符号. 这是因为 NURBS 曲体的边界切向量完全由其边界曲面所决定. 由于样条参数化至少具有 C^1 连续性, 易知其内部至少存在一个雅可比行列式为零的点. 这意味着, 仅使用一块 C^k $(k \geq 1)$ 的 NURBS 体无法实现适合分析的参数化. 关于单值参数化存在性的完整讨论超出了本文范围. 本文中我们假设在给定的边界表示下, 至少存在一个双射的参数化.

2.3.3 各向同性参数化质量指标

如前所述, 计算域参数化的一个基本要求是满足单值性[48]. 除此之外, 一个高质量的参数化还应该具有良好的正交性和均匀性[53,57]. 在一些文献中, 也采用低畸变 (扭曲) 的表述[1,60,114], 本质上它们是一致的.

在定义刻画参数化 $\boldsymbol{x}$ 质量的指标时, 常常涉及到以下基本量:

- 雅可比矩阵:

$$\text{二维情形:} \qquad \mathcal{J} = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix}, \tag{2.20a}$$

$$\text{三维情形:} \qquad \mathcal{J} = \begin{bmatrix} x_\xi & x_\eta & x_\zeta \\ y_\xi & y_\eta & y_\zeta \\ z_\xi & z_\eta & z_\zeta \end{bmatrix}. \tag{2.20b}$$

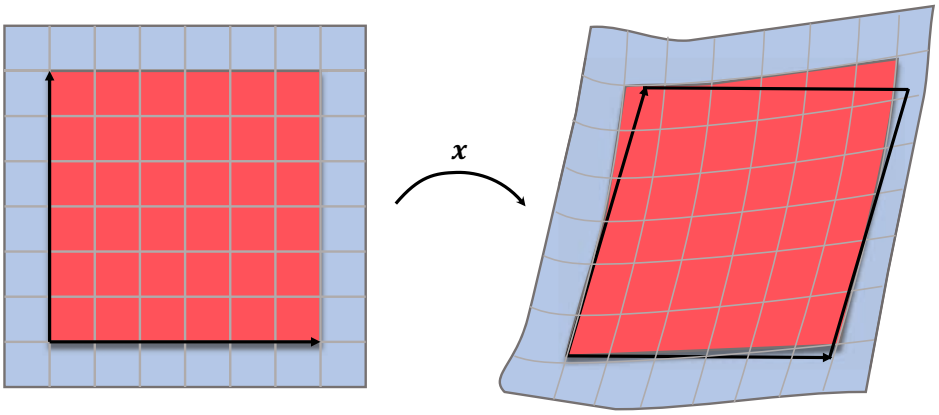


图 2.8 雅可比行列式的几何意义
Fig. 2.8 Geometric Interpretation of Jacobian Determinant

如图 2.8 所示, 雅可比矩阵在一点 $\boldsymbol{\xi}_0$ 的值 $\mathcal{J}(\boldsymbol{\xi}_0)$ 刻画了 $\boldsymbol{x}$ 在该点附近所引起变形的线性逼近, 而其行列式 $|\mathcal{J}(\boldsymbol{\xi}_0)|$ 给出了该近似平行四边形与原始正方形之间

的面积之比. 三维情形中, 雅可比行列式表示在该点附近所引起体积的变化量.

另一个重要的量是雅可比矩阵 $\mathcal{J}$ 的奇异值 σ_i $(i = 1, 2, \ldots, d)$. 如图 2.9 所示, 这些奇异值刻画了局部上, 单位球经过映射后变为椭球的主轴长度的变化情况. 理想情形下, 希望雅可比矩阵的所有奇异值具有相近的大小, 以达到保角的目的.

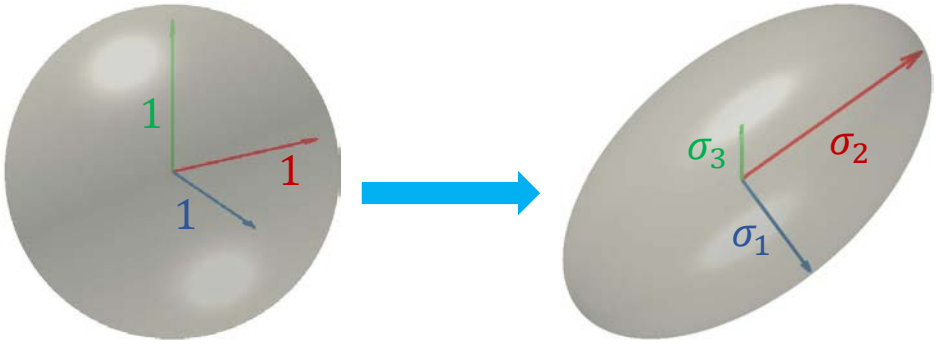


图 2.9　雅可比矩阵奇异值的几何意义

Fig. 2.9　Geometric Interpretation of the Singular Values of the Jacobian Matrix

- 度量张量 $\mathcal{G} = \mathcal{J}^{\mathrm{T}}\mathcal{J}$:

$$\text{二维情形:} \qquad \mathcal{G} = \begin{bmatrix} \boldsymbol{x}_\xi \cdot \boldsymbol{x}_\xi & \boldsymbol{x}_\xi \cdot \boldsymbol{x}_\eta \\ \boldsymbol{x}_\eta \cdot \boldsymbol{x}_\xi & \boldsymbol{x}_\eta \cdot \boldsymbol{x}_\eta \end{bmatrix}, \tag{2.21a}$$

$$\text{三维情形:} \qquad \mathcal{G} = \begin{bmatrix} \boldsymbol{x}_\xi \cdot \boldsymbol{x}_\xi & \boldsymbol{x}_\xi \cdot \boldsymbol{x}_\eta & \boldsymbol{x}_\xi \cdot \boldsymbol{x}_\zeta \\ \boldsymbol{x}_\eta \cdot \boldsymbol{x}_\xi & \boldsymbol{x}_\eta \cdot \boldsymbol{x}_\eta & \boldsymbol{x}_\eta \cdot \boldsymbol{x}_\zeta \\ \boldsymbol{x}_\zeta \cdot \boldsymbol{x}_\xi & \boldsymbol{x}_\zeta \cdot \boldsymbol{x}_\eta & \boldsymbol{x}_\zeta \cdot \boldsymbol{x}_\zeta \end{bmatrix}. \tag{2.21b}$$

特别地, 参数化 $\boldsymbol{x}$ 在某一点 (ξ_0, η_0) 附近是局部保形参数化, 当且仅当 $\mathcal{G} = n(\xi_0, \eta_0) \cdot \mathbf{I}_{d\times d}$, 其中 $\mathbf{I}_{d\times d}$ 是单位阵, 且 $n(\xi_0, \eta_0) > 0$. 此时, 雅可比矩阵 $\mathcal{J}$ 仅是每个方向同比例缩放和旋转的乘积, 参数化 $\boldsymbol{x}$ 具有正交的等参结构. 注意, 这时雅可比矩阵 $\mathcal{J}$ 的所有奇异值都相同.

在本论文中, 我们采用如下的指标度量各向同性参数化结果的质量:

- 比例雅可比:

$$\text{二维情形:} \qquad |\mathcal{J}|_s = \frac{|\mathcal{J}|}{\|\boldsymbol{x}_\xi\|\|\boldsymbol{x}_\eta\|}, \tag{2.22a}$$

$$\text{三维情形:} \qquad |\mathcal{J}|_s = \frac{|\mathcal{J}|}{\|\boldsymbol{x}_\xi\|\|\boldsymbol{x}_\eta\|\|\boldsymbol{x}_\zeta\|}. \tag{2.22b}$$

比例雅可比 $|\mathcal{J}|_s$ 刻画了参数化等参结构的正交性, 且有 $|\mathcal{J}|_s \in [-1, 1]$. 参数化 $\boldsymbol{x}$ 是单值的当且仅当 $|\mathcal{J}|_s > 0, \forall \boldsymbol{\xi} \in \hat{\Omega}$. 只要比例雅可比 $|\mathcal{J}|_s$ 出现了负值, 则说明

该参数化不是单值的. 如果在整个参数域上比例雅可比 $|\boldsymbol{\mathcal{J}}|_s$ 的值接近 1.0, 则说明参数化具有良好的正交性.

- 雅可比矩阵的 Frobenius 条件数:

$$\text{cond}(\boldsymbol{\mathcal{J}}) = \|\boldsymbol{\mathcal{J}}\|_F \|\boldsymbol{\mathcal{J}}\|_F^{-1}, \tag{2.23}$$

其中 $\|\boldsymbol{\mathcal{J}}\|_F = (\text{trace}(\boldsymbol{\mathcal{J}}^{\text{T}}\boldsymbol{\mathcal{J}}))^{\frac{1}{2}}$ 是雅可比矩阵 $\boldsymbol{\mathcal{J}}$ 的 Frobenius 范数. 这个指标刻画了参数化 $\boldsymbol{x}$ 的角度扭曲. 容易得知 cond($\boldsymbol{\mathcal{J}}$) 的最小值为 2.0, 所以如果该指标在整个参数域上均接近 2.0, 则说明该参数化具有较低的角度扭曲.

- 均匀性指标:

$$m^{\text{unif.}} = \left(\frac{|\boldsymbol{\mathcal{J}}|}{area(\Omega)} - 1\right)^2, \tag{2.24}$$

这里 $area(\Omega)$ 表示计算域 Ω 的面积 (或体积). 理想情况下, 参数化 $\boldsymbol{x}$ 雅可比行列式的值应该等于计算域的面积 (或体积) 与参数域的面积 (或体积) 的比值. 因此, 均匀性指标 $m^{\text{unif.}}$ 的最佳值是 0.

3 基于障碍函数的高质量平面参数化方法

3.1 引言

现代 CAD 系统通常仅提供模型的边界表示, 而在等几何分析中, 需要如图 3.1 所示的内部参数表示. 从模型的边界表示出发构建计算域参数化是等几何分析的基本问题和关键步骤. 大量研究表明参数化的质量对后续分析的精度和计算效率具有显著影响[48–50]. 在这里, 高质量的参数化是指在保证单值性的前提下, 等参线具有良好的正交性和均匀性. 文献[1,78,114]中所提到的低扭曲 (low distortion) 概念具有相似的含义. 本章重点关注构造高质量且适合分析的平面参数化方法. 需要特别指出的是, 本方法完全适用于三维体参数化情形.

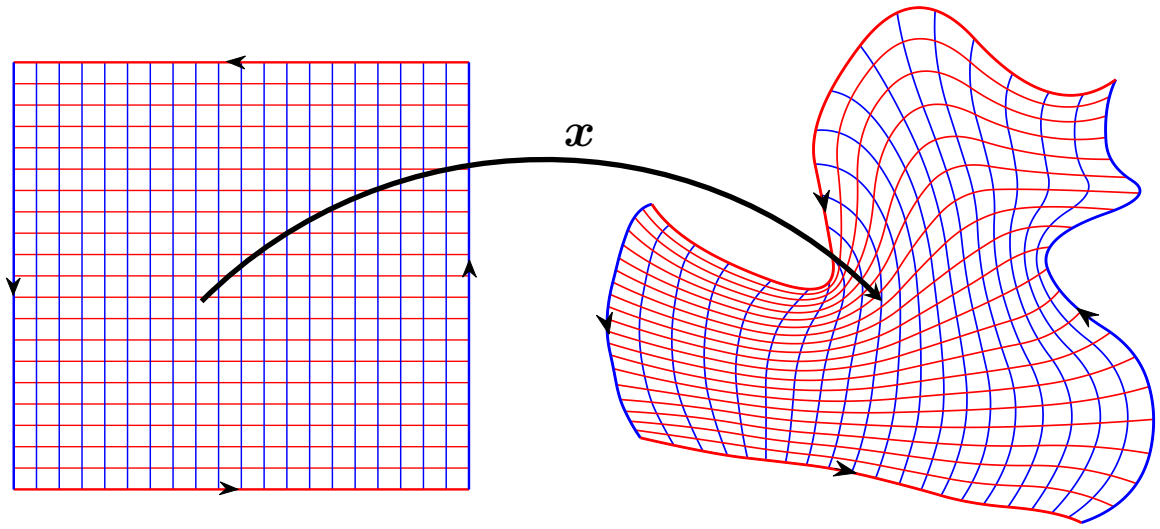


图 3.1 区域参数化: 从边界表示构造参数化 $\boldsymbol{x}$

Fig. 3.1 Domain Parameterization: Constructing Parameterization $\boldsymbol{x}$ from Boundary Representation

对于适合分析的参数化, 保证其单值性至关重要. 现有文献主要讨论 B 样条参数化的单值性条件, 如[49,53,54,56], 但这些条件并不适用于更一般的 NURBS 参数化. 然而, 对 NURBS 参数化的单值性讨论尤为重要. 这是因为在几何建模中, 一些常见的圆锥曲线和曲面 (如椭圆, 双曲线, 圆柱和球面) 无法用 B 样条精确表示. 为此, 本章首先详细地探讨了 NURBS 参数化单值性的若干必要条件和充分条件, 并基于这些条件提出了一个实用的 NURBS 参数化单值性检验算法.

同时, 我们知道 NURBS 的几何形状可以通过调整控制顶点、权因子和节点向量来修改. 因此, 同样可以通过这三种方法来提高计算域参数化的质量. 由于节点向量由

输入的边界表示所确定, 为了保证计算域的几何精确性, 节点向量应当保持不变. 本章选择将内部控制点和权因子作为优化变量, 并提出了一种高质量且鲁棒的 NURBS 参数化方法, 即障碍函数方法. 为了提高计算效率, 我们设计了一种避免计算和更新雅可比行列式系数的算法, 并将消除折叠和参数化质量提升分开处理, 实现了高效的参数化构造. 与现有的方法相比, 本章方法展现出了更好的计算效率和参数化质量.

3.2 单值性条件和检验算法

在等几何分析中, 一个适合分析的参数化首先需要保证其具有双射 (一一映射) 性质. 由于 CAD 产品的边界几何限制, 实现满射 (满足映射到整个目标集合) 通常相对容易. 因此, 大量的研究聚焦于如何构建满足单值性条件的参数化方法. 然而, 在一般 NURBS 参数化情形下, 寻找满足单射性质的充分必要条件是一项极具挑战性的任务. 针对这一问题, 本节专门研究了平面 NURBS 参数化的单值性, 提出了多个充分条件和一个必要条件. 基于这些条件, 我们进一步设计了一个实用的算法, 用于验证平面 NURBS 参数化是否满足单值性. 这为处理等几何分析中的参数化问题提供了有力的理论支持和实用工具.

3.2.1 NURBS 参数化单值性的充分条件

本章假定计算域 $\Omega \subset \mathbb{R}^2$ 是单连通的, 即 Ω 的边界 $\partial\Omega$ 由 4 条具有至少 C^0 连续的 NURBS 曲线所定义. 给定的边界曲线无自交, 并且仅在其端点上成对相交. 此外, 我们假设给定的边界 NURBS 曲线定义在如下两个开的节点向量

$$\boldsymbol{\Xi} = \left\{ \underbrace{0,0,\ldots,0}_{p+1}, \underbrace{\xi_1,\xi_1,\ldots,\xi_1}_{m_1^\xi}, \ldots, \underbrace{\xi_i,\xi_i,\ldots,\xi_i}_{m_i^\xi}, \ldots, \underbrace{1,1,\ldots,1}_{p+1} \right\}$$

和

$$\boldsymbol{\mathcal{H}} = \left\{ \underbrace{0,0,\ldots,0}_{q+1}, \underbrace{\eta_1,\eta_1,\ldots,\eta_1}_{m_1^\eta}, \ldots, \underbrace{\eta_i,\eta_i,\ldots,\eta_i}_{m_i^\eta}, \ldots, \underbrace{1,1,\ldots,1}_{q+1} \right\}$$

上, 其中 m_i^ξ 和 m_i^η 分别是 ξ-和 η-参数方向上内部节点值的重数.

一般地, 一个平面 NURBS 参数化具有如下的数学表示

$$\begin{aligned} \boldsymbol{x}: \quad \hat{\Omega} \quad &\to \quad \Omega \subset \mathbb{R}^2 \\ (\xi,\eta)^{\mathrm{T}} \mapsto (x,y)^{\mathrm{T}} &= \frac{\sum\limits_{i=0}^{n}\sum\limits_{j=0}^{m} \omega_{ij}\mathbf{P}_{ij}N_i^p(\xi)N_j^q(\eta)}{\sum\limits_{i=0}^{n}\sum\limits_{j=0}^{m} \omega_{ij}N_i^p(\xi)N_j^q(\eta)}, \end{aligned} \tag{3.1}$$

其中 $\mathbf{P}_{ij}=(x_{ij},\ y_{ij})^{\mathrm{T}}\in\mathbb{R}^2$ 是控制顶点, ω_{ij} 是相应的权值, $N_i^p(\xi)$ 和 $N_j^q(\eta)$ 分别是 p 和 q 次 B 样条基函数.

由于上式 (3.1) 中的 NURBS 参数化 $\boldsymbol{x}$ 是一个有理映射, 其偏导数的计算相较于 B 样条参数化情形更为复杂, 我们有

$$\begin{aligned}\frac{\partial\boldsymbol{x}}{\partial\xi}&=\frac{p}{W^2}\left(\mathbf{P}_{,\xi}^{\omega}W-\mathbf{P}^{\omega}W_{,\xi}\right),\\ \frac{\partial\boldsymbol{x}}{\partial\eta}&=\frac{q}{W^2}\left(\mathbf{P}_{,\eta}^{\omega}W-\mathbf{P}^{\omega}W_{,\eta}\right),\end{aligned}\tag{3.2}$$

其中

$$W=\sum_{i=0}^{n}\sum_{j=0}^{m}N_i^p(\xi)N_j^q(\eta)\omega_{ij},\quad \mathbf{P}^{\omega}=\sum_{i=0}^{n}\sum_{j=0}^{m}N_i^p(\xi)N_j^q(\eta)\omega_{ij}\mathbf{P}_{i,j},$$

$$W_{,\xi}=\sum_{i=0}^{n-1}\sum_{j=0}^{m}N_i^{p-1}(\xi)N_j^q(\eta)\alpha_i\Delta\omega_{ij,\xi},\quad W_{,\eta}=\sum_{i=0}^{n}\sum_{j=0}^{m-1}N_i^p(\xi)N_j^{q-1}(\eta)\beta_j\Delta\omega_{ij,\eta},$$

$$\mathbf{P}_{,\xi}^{\omega}=\sum_{i=0}^{n-1}\sum_{j=0}^{m}N_i^{p-1}(\xi)N_j^q(\eta)\alpha_i\Delta^{\omega}\mathbf{P}_{ij,\xi},\quad \mathbf{P}_{,\eta}^{\omega}=\sum_{i=0}^{n}\sum_{j=0}^{m-1}N_i^p(\xi)N_j^{q-1}(\eta)\beta_j\Delta^{\omega}\mathbf{P}_{ij,\eta},$$

$$\Delta^{\omega}\mathbf{P}_{ij,\xi}=\omega_{i+1,j}\mathbf{P}_{i+1,j}-\omega_{i,j}\mathbf{P}_{i,j},\quad \Delta^{\omega}\mathbf{P}_{ij,\eta}=\omega_{i,j+1}\mathbf{P}_{i,j+1}-\omega_{i,j}\mathbf{P}_{i,j},$$

$$\Delta\omega_{ij,\xi}=\omega_{i+1,j}-\omega_{i,j},\quad \Delta\omega_{ij,\eta}=\omega_{i,j+1}-\omega_{i,j},$$

$$\alpha_i=\frac{1}{\xi_{i+p+1}-\xi_{i+1}},\quad \beta_j=\frac{1}{\eta_{j+q+1}-\eta_{j+1}},$$

ξ_i 和 η_j 分别是节点向量 $\boldsymbol{\Xi}$ 和 $\boldsymbol{\mathcal{H}}$ 中第 i 个和第 j 个节点值.

利用 B 样条的乘积性质[115], 可以将参数化 $\boldsymbol{x}$ 的雅可比行列式 $|\boldsymbol{\mathcal{J}}|$ 表示为

$$\begin{aligned}|\boldsymbol{\mathcal{J}}|&=\left|\frac{\partial\boldsymbol{x}}{\partial\xi},\ \frac{\partial\boldsymbol{x}}{\partial\eta}\right|\\ &=\frac{pq}{W^4}\sum_{i=0}^{n^{\mathcal{J}}}\sum_{j=0}^{m^{\mathcal{J}}}N_i^{4p-1}(\xi)N_j^{4q-1}(\eta)\mathcal{J}_{i,j}^{N},\end{aligned}\tag{3.3}$$

其中 $\mathcal{J}_{i,j}^{N}$ 是 B 样条乘积的控制系数, $n^{\mathcal{J}}=3n^{\xi}p-2$ 和 $m^{\mathcal{J}}=3m^{\eta}q-2$ 分别是 ξ 和 η 参数方向上的控制系数的个数, n^{ξ} 和 m^{η} 分别是 ξ 和 η 参数方向上单元的个数, $N_i^{4p-1}(\xi)$, $N_j^{4q-1}(\eta)$ 是 $4p-1$ 和 $4q-1$ 次 B 样条基函数, 分别定义在节点向量

$$\widetilde{\boldsymbol{\Xi}}=\left\{\underbrace{0,0,\ldots,0}_{4p},\underbrace{\xi_1,\xi_1,\ldots,\xi_1}_{3p+m_1^{\xi}},\ldots,\underbrace{\xi_i,\xi_i,\ldots,\xi_i}_{3p+m_i^{\xi}},\ldots,\underbrace{1,1,\ldots,1}_{4p}\right\}$$

和

$$\widetilde{\boldsymbol{\mathcal{H}}} = \left\{ \underbrace{0,0,\ldots,0}_{4q}, \underbrace{\eta_1,\eta_1,\ldots,\eta_1}_{3q+m_1^\eta}, \ldots, \underbrace{\eta_i,\eta_i,\ldots,\eta_i}_{3q+m_i^\eta}, \ldots, \underbrace{1,1,\ldots,1}_{4q} \right\}$$

上.

式 (3.3) 中每个控制系数 $\mathcal{J}_{i,j}^N$ 是参数化 $\boldsymbol{x}$ 的控制点及权因子的非线性组合. 由于对 $\forall(\xi,\ \eta) \in \hat{\Omega}$, 我们总是有 $pq/W^4 > 0$, 因此雅可比行列式 $|\boldsymbol{\mathcal{J}}|$ 的符号完全由

$$|\mathcal{J}_D(\xi,\eta)| = \sum_{i=0}^{n^{\mathcal{J}}} \sum_{j=0}^{m^{\mathcal{J}}} N_i^{4p-1}(\xi) N_j^{4q-1}(\eta) \mathcal{J}_{i,j}^N \tag{3.4}$$

的符号所决定.

根据 B 样条基函数的非负性质和引理2.1, 我们可以很容易得到下面的定理.

定理 3.1. 对于一个平面 NURBS 参数化 $\boldsymbol{x}$ 而言, 如果式 (3.4) 中的所有控制系数 $\mathcal{J}_{i,j}^N$ 均为正, 那么该 NURBS 参数化 $\boldsymbol{x}$ 是单值的.

定理3.1为检验平面 NURBS 参数化的单值性提供了一个充分条件. 当所有权值相等时, NURBS 参数化 $\boldsymbol{x}$ 退化成 B 样条参数化. 此时, B 样条参数化 $\boldsymbol{x}$ 的雅可比行列式的次数退化为 $(2p-1)(2q-1)$. 这一观察有助于减少计算量. 同时, 这也说明定理3.1的内容包含了[53,54,56,116]中关于 B 样条参数化单值性的结果.

3.2.2 弱化的单值性条件

定理3.1的结论指出: 如果一个 NURBS 参数化 $\boldsymbol{x}$ 的雅可比行列式控制系数 $\mathcal{J}_{i,j}^N$ 的最小值为正, 那么该 NURBS 参数化具有单值性. 然而, 一般的 NURBS 曲面不满足 Kronecker-Delta 性质, 无法插值于所有的控制顶点. 这导致该定理中关于雅可比行列式下界的估计可能过于粗糙, 在很多情况下无法完全满足实际应用的需求.

为解决这一问题, 我们进一步考虑了利用 NURBS 几何的节点插入或升阶算法来寻找更宽松的充分条件, 以确保 NURBS 参数化的单值性. 这是因为在扩展后的节点向量所构成的样条空间中, 新的控制系数是原控制系数 $\mathcal{J}_{i,j}^N$ 的凸组合. 从而可以得到参数化雅可比行列式的一个更紧的下界估计.

相较于前述方法, 一个更为巧妙的方法是利用 Bézier 提取技术将 NURBS 参数化 $\boldsymbol{x}$ 转换为分片有理 Bézier 表示. 具体来说, 即不断地向原始节点向量 $\boldsymbol{\Xi}$ 和 $\boldsymbol{\mathcal{H}}$ 中插入已有的节点值, 直到所有内部节点的重数分别达到 p 和 q. 随后, 对于每一个非空的节点区间, 可以通过伸缩变换将参数域映射到单位正方形上, 那么在每个单元上都可以得到相应的有理 Bézier 表示:

$$\boldsymbol{x}_B(\xi,\eta)=\frac{\sum\limits_{i=0}^{p}\sum\limits_{j=0}^{q}\omega_{ij}\mathbf{P}_{ij}B_i^p(\xi)B_j^q(\eta)}{\sum\limits_{i=0}^{p}\sum\limits_{j=0}^{q}\omega_{ij}B_i^p(\xi)B_j^q(\eta)},\quad (\xi,\eta)\in[0,1]^2, \tag{3.5}$$

其中 $B_i^p(\xi)$ 和 $B_i^q(\eta)$ 是 Bernstein 基函数.

这种做法的第一个好处是计算简单. 与计算 NURBS 表示的乘积相比, 计算 Bézier 表示的乘积要相对容易很多. 根据 Bernstein 基函数的定义可知, 两个 Bernstein 基函数 $B_i^p(\xi)$ 和 $B_j^q(\xi)$ 的乘积为

$$B_i^p(\xi)B_j^q(\xi)=\frac{\binom{p}{i}\binom{q}{j}}{\binom{p+q}{i+j}}B_{i+j}^{p+q}(\xi). \tag{3.6}$$

利用式 (3.6) 中 Bernstein 基函数的乘积性质, 可以按照如下方式计算有理 Bézier 表示 (3.5) 的偏导数:

$$\begin{aligned}
\frac{\partial\boldsymbol{x}_B}{\partial\xi}&=\frac{p}{W_B^2}\left(W_B\sum_{i=0}^{p-1}\sum_{j=0}^{q}\Delta^{\omega}\mathbf{P}_{ij,\xi}B_i^{p-1}(\xi)B_j^q(\eta)-\mathbf{P}_B^{\omega}\sum_{i=0}^{p-1}\sum_{j=0}^{q}\Delta\omega_{ij,\xi}B_i^{p-1}(\xi)B_j^q(\eta)\right)\\
&=\frac{p}{W_B^2}\sum_{\bar{s}_1=0}^{2p-1}\sum_{\bar{s}_2=0}^{2q}\bar{C}^{\xi}_{\bar{s}_1,\bar{s}_2}B_{\bar{s}_1}^{2p-1}(\xi)B_{\bar{s}_2}^{2q}(\eta),\\
\frac{\partial\boldsymbol{x}_B}{\partial\eta}&=\frac{q}{W_B^2}\left(W_B\sum_{i=0}^{p}\sum_{j=0}^{q-1}\Delta^{\omega}\mathbf{P}_{ij,\eta}B_i^{p}(\xi)B_j^{q-1}(\eta)-\mathbf{P}_B^{\omega}\sum_{i=0}^{p}\sum_{j=0}^{q-1}\Delta\omega_{ij,\eta}B_i^{p}(\xi)B_j^{q-1}(\eta)\right)\\
&=\frac{q}{W_B^2}\sum_{\bar{s}_3=0}^{2p}\sum_{\bar{s}_4=0}^{2q-1}\bar{C}^{\eta}_{\bar{s}_3,\bar{s}_4}B_{\bar{s}_3}^{2p}(\xi)B_{\bar{s}_4}^{2q-1}(\eta),
\end{aligned} \tag{3.7}$$

其中

$$W_B=\sum_{i=0}^{p}\sum_{j=0}^{q}B_i^p(\xi)B_j^q(\eta)\omega_{ij},\quad \mathbf{P}_B^{\omega}=\sum_{i=0}^{p}\sum_{j=0}^{q}B_i^p(\xi)B_j^q(\eta)\omega_{ij}\mathbf{P}_{i,j},$$

$$\bar{C}^{\xi}_{\bar{s}_1,\bar{s}_2}=\sum_{\substack{i_1+i_2=\bar{s}_1\\0\leqslant i_1\leqslant p-1\\0\leqslant i_2\leqslant p}}\sum_{\substack{j_1+j_2=\bar{s}_2\\0\leqslant j_1\leqslant q\\0\leqslant j_2\leqslant q}}\frac{\binom{p-1}{i_1}\binom{p}{i_2}\binom{q}{j_1}\binom{q}{j_2}}{\binom{2p-1}{\bar{s}_1}\binom{2q}{\bar{s}_2}}\mathbf{\Gamma}^{\xi}_{i_1,i_2,j_1,j_2},$$

$$\bar{C}^{\eta}_{\bar{s}_3,\bar{s}_4}=\sum_{\substack{i_3+i_4=\bar{s}_3\\0\leqslant i_3\leqslant p\\0\leqslant i_4\leqslant p}}\sum_{\substack{j_3+j_4=\bar{s}_4\\0\leqslant j_3\leqslant q-1\\0\leqslant j_4\leqslant q}}\frac{\binom{p}{i_3}\binom{p}{i_4}\binom{q-1}{j_3}\binom{q}{j_4}}{\binom{2p}{\bar{s}_3}\binom{2q-1}{\bar{s}_4}}\mathbf{\Gamma}^{\eta}_{i_3,i_4,j_3,j_4},$$

$$\mathbf{\Gamma}^{\xi}_{i_1,i_2,j_1,j_2}=\omega_{i_2,j_2}\left(\omega_{i_1+1,j_1}(\mathbf{P}_{i_1+1,j_1}-\mathbf{P}_{i_2,j_2})-\omega_{i_1,j_1}(\mathbf{P}_{i_1,j_1}-\mathbf{P}_{i_2,j_2})\right),$$

且

$$\mathbf{\Gamma}^{\eta}_{i_3,i_4,j_3,j_4} = \omega_{i_4,j_4}\left(\omega_{i_3,j_3+1}(\mathbf{P}_{i_3,j_3+1} - \mathbf{P}_{i_4,j_4}) - \omega_{i_3,j_3}(\mathbf{P}_{i_3,j_3} - \mathbf{P}_{i_4,j_4})\right).$$

由 Bernstein 基函数的乘积性质 (3.6) 和偏导数 (3.7), 在每一个单元上, 我们可以计算得出式 (3.5) 有理 Bézier 表示 $\boldsymbol{x}_B$ 雅可比行列式的 Bézier 控制系数如下

$$\left|\mathcal{J}^B_D(\xi,\eta)\right| = \sum_{s_1=0}^{4p-1}\sum_{s_2=0}^{4q-1} B^{4p-1}_{s_1}(\xi)B^{4q-1}_{s_2}(\eta)\mathcal{J}^B_{s_1,s_2}, \tag{3.8}$$

其中

$$\mathcal{J}^B_{s_1,s_2} = \sum_{\substack{i_1+i_2+i_3+i_4=s_1\\ 0\leqslant i_1\leqslant p-1\\ 0\leqslant i_2\leqslant p\\ 0\leqslant i_3\leqslant p\\ 0\leqslant i_4\leqslant p}} \sum_{\substack{j_1+j_2+j_3+j_4=s_2\\ 0\leqslant j_1\leqslant q\\ 0\leqslant j_2\leqslant q\\ 0\leqslant j_3\leqslant q-1\\ 0\leqslant j_4\leqslant q}} \gamma_{s_1,s_2}\left|\mathbf{\Gamma}^{\xi}_{i_1,i_2,j_1,j_2}, \mathbf{\Gamma}^{\eta}_{i_3,i_4,j_3,j_4}\right|, \tag{3.9}$$

且

$$\gamma_{s_1,s_2} = \frac{\binom{p-1}{i_1}\binom{p}{i_2}\binom{p}{i_3}\binom{p}{i_4}\binom{q-1}{j_1}\binom{q}{j_2}\binom{q}{j_3}\binom{q}{j_4}}{\binom{4p-1}{s_1}\binom{4q-1}{s_2}}.$$

如果所有 Bézier 片的控制系数均大于零, 那么原 NURBS 参数化 (3.1) 是单值的. 因此, 由引理2.1和式 (3.8), 可以很容易地得到 NURBS 参数化单值性更弱的充分条件.

定理 3.2. 记 $\{\boldsymbol{x}^{(i)}_B\}_{i=1}^{n^\xi m^\eta}$ 是平面 NURBS 参数化 $\boldsymbol{x}$ 经 Bézier 提取后得到的 Bézier 表示. 如果对所有的 $i = 1, 2, \ldots, n^\xi m^\eta$, Bézier 表示 $\boldsymbol{x}^{(i)}_B$ 的所有 Bézier 控制系数 $J^B_{s_1,s_2}$ 均大于零, 则平面 NURBS 参数化 $\boldsymbol{x}$ 是单值的.

使用 Bézier 表示 (3.5) 的另一个好处是: 由于 Bézier 表示插值其角点处的控制系数 (或控制顶点), 我们可以得到参数化单值性的一个必要条件.

定理 3.3. 记 $\{\boldsymbol{x}^{(i)}_B\}_{i=1}^{n^\xi m^\eta}$ 是平面 NURBS 参数化 $\boldsymbol{x}$ 经 Bézier 提取后得到的 Bézier 表示. 如果 NURBS 参数化 $\boldsymbol{x}$ 是单值的, 那么对所有的 $i = 1, 2, \ldots, n^\xi m^\eta$, Bézier 表示 $\boldsymbol{x}^{(i)}_B$ 的所有位于角点处的 Bézier 控制系数 $\mathcal{J}^B_{0,0}$, $\mathcal{J}^B_{0,4q-1}$, $\mathcal{J}^B_{4p-1,0}$, 和 $\mathcal{J}^B_{4p-1,4q-1}$ 非负.

注 3.1. 对于具有开节点向量的 NURBS 参数化而言, 其边界曲线完全由边界上的控制顶点和权因子决定. 因此, 内部的控制顶点和权因子不影响计算域边界上的 Bézier 系数. 另外, 相邻 Bézier 片之间的 Bézier 控制系数 $\mathcal{J}^B_{s_1,s_2}$ 是重复的. 基于这些观察, 我们可以通过避免重复计算来减轻计算负担.

3.2.3 单值性检验算法

结合前两个小节提出的充分条件和必要条件, 这一小节对一般的平面 NURBS 参数化给出一种实用的单值性检验算法. 整个算法流程如算法 3.1 所示.

Algorithm 3.1: 平面 NURBS 参数化的单值性检验算法

Input: $\mathbf{P}_{ij}$: 平面 NURBS 参数化 $\boldsymbol{x}$ 的控制顶点;
ω_{ij}: 平面 NURBS 参数化 $\boldsymbol{x}$ 的权因子;
$\Xi, \mathcal{H}$: 平面 NURBS 参数化 $\boldsymbol{x}$ 的节点向量;
p, q: 平面 NURBS 参数化 $\boldsymbol{x}$ 的次数;
max_seg: 最大细化次数.

Output: 平面 NURBS 参数化 $\boldsymbol{x}$ 的单值性.

```
计算每个参数方向上的单元数量 n^ξ, m^η;
利用 Bézier 提取计算参数化 x 的 Bézier 片 {x_B^(i)}_{i=1}^{n^ξ m^η};
在列表 List_marked 中存储 {x_B^(i)}_{i=1}^{n^ξ m^η};
while k=0; k ≤ max_seg; k++ do
    for 列表 List_marked 中每个 Bézier 片 do
        计算雅可比行列式 |J_D^B(ξ,η)| 的 Bézier 控制系数 J_{s1,s2}^B;    // (3.9)
        // 平面 NURBS 参数化单值性的必要条件: 定理3.3
        if min{J_{0,0}^B, J_{0,4q-1}^B, J_{4p-1,0}^B, J_{4p-1,4q-1}^B} ≤ 0 then
            return 平面 NURBS 参数化 x 不单值;
        end
        if min{J_{s1,s2}^B}_{s1=0,s2=0}^{4p-1,4q-1} < 0 then
            利用 de Casteljau 算法将该 Bézier 片分割成 4 个子 Bézier 片;
            在列表 List_marked 中添加这 4 个子 Bézier 片;
        end
        在列表 List_marked 中删除当前 Bézier 片;
    end
    if 列表 List_marked 为空 then
        break;
    end
end
if 列表 List_marked 非空 then
    return "已达到最大分割次数, 无法确定 x 的单值性!";
end
return "平面 NURBS 参数化 x 单值";
```

如前所述, 在控制顶点数量较少时, 前文介绍的充分条件可能不足以检验某些 NURBS 参数化的单值性. 为解决这一问题, 我们利用 Bézier 表示的 de Casteljau 算法[7] 将 Bézier 表示 (3.8) 分割成四个子片. 通过这种方式, 我们可以推导出一系列更弱的充分条件和一系列必要条件. 此时, 分割后的 Bézier 控制系数二次收敛至原始的 $|\mathcal{J}_D(\xi,\eta)|$. 这是因为样条函数的控制多边形二次收敛于该样条函数本身. 有关这一性质的证明, 感兴趣的读者可以参考文献[117]第 197 页的证明.

需要注意的是, 更弱的充分条件也伴随着更高的计算负担. 不过, 只有带有非正 Bézier 控制系数的 Bézier 片 (通常只占一小部分) 需要进行分割. 因此, 算法 3.1 能够在

可接受的时间内完成计算. 在我们的实现中, 最大分割次数 max_seg 被设定为 5. 对于本章中涉及的所有模型, 除了兔子模型需要进行 3 次分割之外, 无需额外的分割便足以检验其他模型的单值性.

3.3 基于障碍函数的参数化方法与交替更新策略

在本节中, 我们讨论一种构造高质量平面 NURBS 参数化的障碍函数方法. 在这里, 我们不仅考虑内部控制顶点对参数化质量的影响, 还考虑了内部权因子的作用.

对于面向等几何分析的适合分析的参数化构造, 一般地, 基于约束优化的框架可以归结为如下的数学模型:

$$\begin{aligned} \underset{var}{\arg\min} \quad & \mathscr{E}\left(\boldsymbol{x}(var)\right) \\ \text{s.t.} \quad & \mathcal{J}_{s_1,s_2} \geq \delta_0 > 0, \end{aligned}$$

其中 var 是优化变量, 可以取为内部控制顶点或权因子, $\mathscr{E}$ 是刻画参数化 $\boldsymbol{x}$ 几何特性的能量泛函, 如正交性和均匀性[49,53], 或者是低扭曲[1,2,114], $\mathcal{J}_{s_1,s_2}$ 是式 (3.4) 或式 (3.8) 中雅可比行列式的控制系数, $\delta_0 > 0$ 是一个正的参数, 一般取为一个很小的正值.

然而, 对于一个双三次 NURBS 参数化而言, 假设其每个参数方向上有 20 个控制顶点且其所有的内部节点值重数均为 1, 那么式 (3.8) 中 Bézier 系数的数量达 34596 之多. 因此, 当 CAD 模型的控制顶点数目较多时, 计算控制系数 $\mathcal{J}_{s_1,s_2}$ 会消耗大量时间和内存. 此外, 在利用上述约束优化方法 (3.3) 构造适合分析的参数化时, 由于在每步迭代中都需要更新这些 Bézier 系数, 故将这些条件作为目标函数或约束的一部分仍然不是一个明智的选择. 所以, 在大规模应用中直接求解上述约束优化问题在计算上是不可接受的. 为此, 本节提出一种无约束优化方法来高效地构造适合等几何分析的高质量参数化. 我们称之为障碍函数方法.

3.3.1 第一步: 初始参数化构造

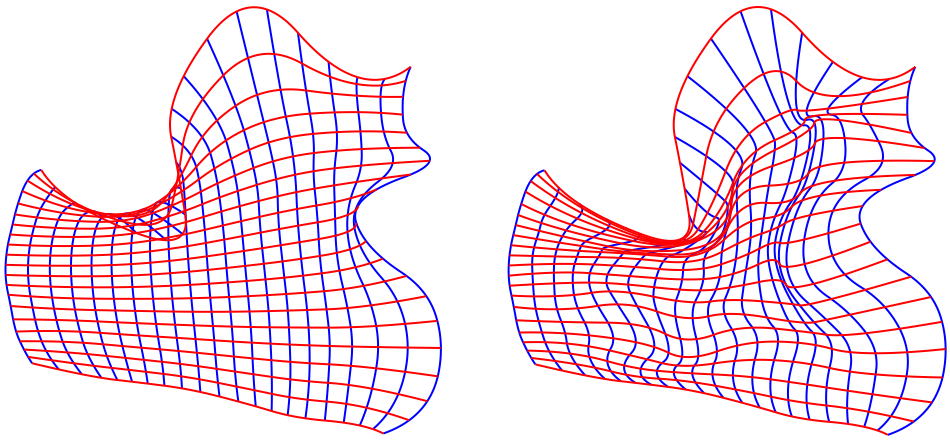
求解非线性优化问题的一般方法是迭代方法, 因此一个合理的初始猜测是提高后续求解收敛速度的关键. 在等几何分析参数化问题中, 一般采用代数参数化方法生成一个初始猜测, 如离散 Coons 方法[51]、弹簧模型方法[52]、变形方法[54] 和前向 Laplacian 方法[2] 等. 在本章中, 我们采用前向 Laplacian 方法[2] 计算初始参数化.

由于 NURBS 参数化 $\boldsymbol{x}$ 关于权因子是高度非线性的, 在初始化阶段, 我们将所有的内部权因子置为 1. 然后, 通过求解以下无约束二次规划问题得到内部控制点

$$\underset{\mathbf{P}_i,\ i\in\mathcal{I}_I}{\arg\min} \iint_{\hat{\Omega}} \|\Delta_{\boldsymbol{\xi}}\boldsymbol{x}\|^2 \, d\hat{\Omega}, \tag{3.10}$$

其中 $\mathbf{P}_i,\ i \in \mathcal{I}_I$ 是未知的内部控制顶点.

对于上述二次规划问题 (3.10), 可以通过求解一个系数矩阵稀疏且对称的线性系统得到. 通常可以采用不完全 Cholesky 分解的预条件共轭梯度法或 GMRES 方法求解. 由该方法构造的初始参数化如图 3.2(a) 所示. 可以看到, 该方法无法保证得到的参数化没有自交, 在鸭子的背部有许多明显的自交.



(a) 初始化　　(b) 消除折叠

图 3.2　初始化和消除折叠
Fig. 3.2　Initialization and Foldover Elimination

3.3.2　第二步: 消除折叠

一般地, 对复杂的计算域来说, 由前向 Laplacian 方法 (3.10) 所构造的初始参数化并不能保证单值性. 因此, 在我们的方法中, 第二步是消除折叠.

为了构造单值的参数化, 我们需要式 (3.8) 中所有系数 $\mathcal{J}^B_{s_1,s_2}$ 均大于零. 一般地, 可以求解如下的极大极小问题[52]:

$$\arg\max_{\mathbf{P}_i,\ i\in\mathcal{I}_I} \min_{s_1,s_2} \mathcal{J}^B_{s_1,s_2}. \tag{3.11}$$

求解上述问题 (3.11) 可以达到消除折叠的目的. 然而, 该方法在迭代过程中仍无法避免计算雅可比行列式的控制系数 $J^B_{s_1,s_2}$, 导致其计算效率低下. 因此, 我们转而求解如下的无约束优化问题:

$$\arg\min_{\mathbf{P}_i,\ i\in\mathcal{I}_I} \mathscr{E}^{\text{fold}}, \tag{3.12}$$

其中

$$\mathscr{E}^{\text{fold}} = \iint_{\hat{\Omega}} \max\{0,\ \delta - |\mathcal{J}|\}\, \mathrm{d}\hat{\Omega}, \tag{3.13}$$

δ 是用户指定的参数值.

图 3.3　求解无约束问题 (3.12) 得到几乎单值的参数化

Fig. 3.3　Obtaining an Almost Bijective Parameterization by Solving the Unconstrained Problem (3.12)

显然, 上述无约束优化问题 (3.12) 中目标函数的最小值为零. 在实际计算中, 目标函数通常采用 Gauss 数值积分计算. 在这种情况下, 目标函数值为零仅仅意味着被积函数在 Gauss 积分点处为零, 即参数化并不一定是单值的. 我们称这种在所有 Gauss 积分点上满足雅可比行列式大于零的参数化为几乎单值的参数化. 它与满足单值性和一般参数化的关系见图 3.3. 如图 3.2(b) 所示, 通过求解优化问题 (3.12), 大部分的自相交被消除了. 而对绝大多数复杂计算域, 仅求解这一问题并不能完全消除自相交, 不过我们仍可以获得一个几乎单值的参数化. 消除折叠对后续参数化质量提升的成功有至关重要的影响. 第 3.4 节中的表 3.1 中数据表明了消除自相交步骤的有效性和必要性.

Algorithm 3.2: 消除折叠

Input: $\boldsymbol{x}$: 平面 NURBS 参数化;
$decay_factor$: 参数 δ 的衰减因子;
$area(\Omega)$: 计算域的面积;
max_iter: 最大迭代次数.

Output: $\boldsymbol{x}$: 消除折叠后的平面 NURBS 参数化.

```
for k=0; k ≤ max_iter; k++ do
    计算 δ = decay_factor^k 0.05 area(Ω);
    求解无约束优化问题 arg min_{P_ij} ℰ^fold;                    // (3.12)
    更新平面 NURBS 参数化 x 的控制顶点 P_ij;
    // MACHINE_PRECISION: 机器精度
    if ℰ^fold < 100 MACHINE_PRECISION then
        return 消除折叠后的平面 NURBS 参数化 x;
    end
end
return "已达到最大迭代次数!";
```

问题 (3.12) 中, 参数 δ 的选取对问题能否成功求解具有至关重要的影响. 一方面,

我们希望参数 δ 尽可能的大, 这意味着得到的参数化结果质量更高, 有助于后续参数化质量提升的收敛效率; 另一方面, 如果该参数 δ 设置过大, 可能会导致求解失败. 为了增强本章方法的鲁棒性, 我们采取一个从大到小的自适应求解策略. 对于一个一般的参数化 $\boldsymbol{x}$ 而言, 其在某一参数值处的雅可比行列式的值刻画了该点附近从参数域到计算域面积的变化比. 由于我们假设参数域为单位正方形 $[0,1]^2$, 因此其雅可比行列式的值应该等于计算域的面积. 为此, 我们将参数 δ 的初始值设置为计算域面积的 5%. 如果在这一参数值下问题求解失败, 我们将按照一个衰减因子 $decay_factor$ 来逐步减小参数 δ 的值. 具体求解步骤见算法 3.2.

若问题 (3.12) 成功求解, 我们仅能保证在 Gauss 点积分点上的雅可比行列式值大于零, 而不能保证完全消除自相交. 此外, 从图 3.2(b) 中可以看出, 参数化的质量仍然很差. 因此, 接下来需要进一步消除折叠并提高参数化质量.

3.3.3 第三步: 参数化质量提升

粗略来说, 用于描述计算域参数化质量的能量泛函大致可以分为两类: 一类是正交和均匀能量泛函的加权组合[49,53]; 另一类能量泛函刻画参数化的扭曲, 如[1,2,114]. 本质上, 这两类能量泛函的目的是一致的, 均旨在构造具有良好正交性和均匀性的参数化. 在本章中, 我们采用修正的 Winslow 泛函和均匀性泛函的加权和作为目标函数.

修正的 Winslow 泛函. 描述参数化 $\boldsymbol{x}$ 扭曲的 Winslow 泛函[52,62] 定义如下:

$$\begin{aligned}\mathscr{E}^{Winslow}(\boldsymbol{x};\ \mathbf{P}_i,\ \omega_i) &= \iint_{\hat{\Omega}} \frac{trace(\boldsymbol{\mathcal{G}})}{|\boldsymbol{\mathcal{J}}|} d\hat{\Omega} \\ &= \iint_{\hat{\Omega}} \frac{x_\xi^2 + x_\eta^2 + y_\xi^2 + y_\eta^2}{x_\xi y_\eta - y_\xi x_\eta} d\hat{\Omega},\end{aligned} \tag{3.14}$$

其中 $|\boldsymbol{\mathcal{J}}|$ 是雅可比矩阵 $\boldsymbol{\mathcal{J}}$ 的行列式, $trace(\boldsymbol{\mathcal{G}})$ 是第一基本形式 $\boldsymbol{\mathcal{G}}$ 的迹.

Winslow 泛函 $\mathscr{E}^{Winslow}$ 仅与参数化 $\boldsymbol{x}$ 的雅可比行列式及第一基本形式 $\mathcal{G}$ 相关, 因而是内蕴几何量. 由于参数域 $\hat{\Omega} = [0,1]^2$ 是凸的, 依据 Radó-Kneser-Choquet 定理[118–120], 我们可以得知 Winslow 泛函的唯一最小值提供了从参数域内部到计算域内部的微分同胚映射. 此外, 由于雅可比行列式位于分母位置, 当其趋近于零时, Winslow 泛函 $\mathscr{E}^{Winslow}$ 的值将趋向于正无穷. 这有助于抑制自相交的出现.

然而在数值优化中, 由于目标函数并不连续变化, 雅可比行列式的值可能直接越过零. 而一旦这种情况发生, 目标函数值不再有最小值, 此时使用原始的 Winslow 泛函 (3.14) 会导致求解失败. 因此, 我们对原来的 Winslow 泛函做如下修正:

$$\mathscr{E}^{\text{correctedWinslow}}(\boldsymbol{x};\ \mathbf{P}_i,\ \omega_i) = \begin{cases} \mathscr{E}^{Winslow}(\boldsymbol{x};\ \mathbf{P}_i,\ \omega_i), & \text{若} \min|\mathcal{J}| > 0, \\ +\infty, & \text{其他}. \end{cases} \tag{3.15}$$

换言之, 当雅可比行列式在 Gauss 积分点上的最小值小于零时, 原来的 Winslow 泛函泛函被修正为正无穷, 以惩罚无效的参数化. 可以想象, 修正的 Winslow 泛函 $\mathscr{E}^{\text{correctedWinslow}}$ 在单值和非单值参数化之间筑起一道壁垒, 这也是为什么我们称本章的方法为障碍函数方法. 现代的非线性优化求解器往往带有线搜索以保证目标函数值充分的下降, 比如 Armijo-Goldstein 准则, Wolfe-Powell 准则, 强 Wolfe 准则等, 因此修正后的 Winslow 泛函 $\mathscr{E}^{\text{correctedWinslow}}$ 能够有效避免自相交的出现.

均匀性能量泛函. 极小化修正后的 Winslow 泛函 $\mathscr{E}^{\text{correctedWinslow}}$ 能够构造有效的参数化, 结果如图 3.4(a) 所示. 但是该参数化的均匀性并不令人满意, 特别是在边界附近, 如鸭子的肚子和脑袋. 为了克服这一缺陷, 我们考虑在目标泛函中加入均匀性泛函.

根据雅可比行列式的几何意义, 在最均匀的情况下, 雅可比行列式 $|\boldsymbol{\mathcal{J}}|$ 等于计算域的面积 $area(\Omega)$. 因此, 我们定义如下均匀性能量泛函来刻画 NURBS 参数化均匀性:

$$\mathscr{E}^{\text{unif.}}(\boldsymbol{x};\ \mathbf{P}_i,\ \omega_i) = \iint_{\hat{\Omega}} \left(\frac{|\boldsymbol{\mathcal{J}}|}{area(\Omega)} - 1 \right)^2 \mathrm{d}\hat{\Omega}. \tag{3.16}$$

在有了初始参数化后, 计算域的面积 $area(\Omega)$ 可以通过如下公式直接计算

$$area(\Omega) = \iint_{\Omega} 1\ \mathrm{d}\Omega = \iint_{\hat{\Omega}} |\boldsymbol{\mathcal{J}}|\ \mathrm{d}\hat{\Omega}. \tag{3.17}$$

这里, 我们提供一个更高效的方法. 该方法仅用边界 NURBS 曲线来计算 $area(\Omega)$. 利用格林公式[121], 我们有

$$area(\Omega) = \iint_{\Omega} 1\ \mathrm{d}x\mathrm{d}y = \oint_{\partial\Omega} x(t)y^{'}(t)\ \mathrm{d}t = \sum_{i=1}^{4} \int_0^1 x_i(t)y_i^{'}(t)\ \mathrm{d}t, \tag{3.18}$$

其中 $x(t) = \sum\limits_{i=1}^{4} x_i(t)$ 和 $y(t) = \sum\limits_{i=1}^{4} y_i(t)$ 定义了边界 $\partial\Omega$ 的 x-和 y-分量. 在实现中, 需要特别注意此时要求边界曲线逆时针定义整个计算域的边界 $\partial\Omega$.

目标泛函和交替更新策略. 结合修正的 Winslow 泛函 (3.15) 和均匀性泛函 (3.16), 我们构造如下加权目标泛函:

$$\mathscr{E}(\boldsymbol{x};\ \mathbf{P}_i,\ \omega_i) = \mathscr{E}^{\text{correctedWinslow}}(\boldsymbol{x};\ \mathbf{P}_i,\ \omega_i) + \lambda_{\text{unif.}}\mathscr{E}^{\text{unif.}}(\boldsymbol{x};\ \mathbf{P}_i,\ \omega_i), \tag{3.19}$$

其中 $\lambda_{\text{unif.}}$ 是一个正的权值, 用以平衡改进的 Winslow 泛函和均匀性泛函的权重.

本章中, 我们不仅考虑内部控制顶点, 还考虑内部权因子对参数化质量的影响. 为了避免内部控制顶点和权因子的相互影响, 我们采取如下的交替更新策略:

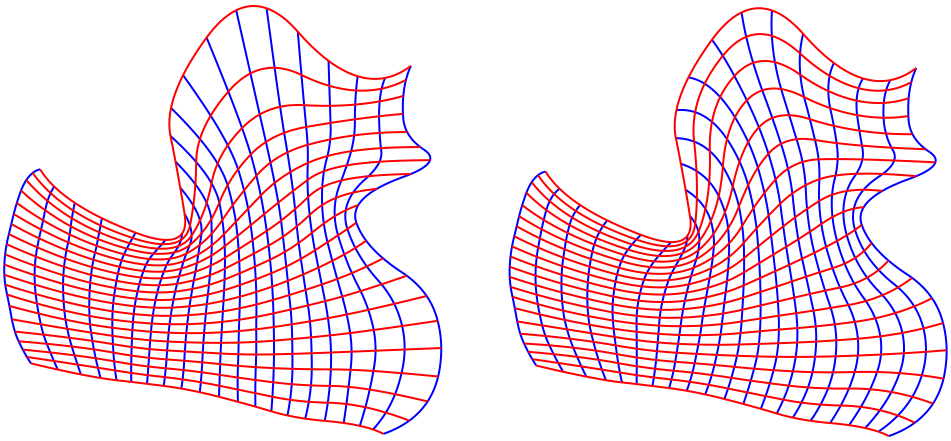
(i) 固定权因子 ω_i^*, 更新内部控制点 $\mathbf{P}_i$. 即求解无约束优化问题:

$$\underset{\mathbf{P}_i,\ i\in\mathcal{I}_I}{\arg\min}\quad \mathscr{E}(\boldsymbol{x};\ \mathbf{P}_i,\ \omega_i^*); \tag{3.20}$$

(ii) 固定内部控制点 $\mathbf{P}_i^*$, 更新内部权因子 ω_i. 即求解无约束优化问题:

$$\underset{\omega_i,\ i\in\mathcal{I}_I}{\arg\min}\quad \mathscr{E}(\boldsymbol{x};\ \mathbf{P}_i^*,\ \omega_i). \tag{3.21}$$

在我们的实现中, 交替求解上述两个子问题, 直到更新控制顶点后的目标函数值 $\mathscr{E}^{pt}$ 与更新权因子后的目标函数值 $\mathscr{E}^{wt}$ 的相对变化量 $|\mathscr{E}^{pt}-\mathscr{E}^{wt}|$ 小于用户指定的阈值 ε_{rel}, 或是达到最大迭代次数 $max_iter_alternating$. 在我们的实验中, 一般经过 1–2 次迭代即可收敛. 通过交替求解上面两个子问题, 生成的参数化如图 3.4(b) 所示. 可以看出, 在均匀性泛函 (3.16) 的作用下, 参数化的均匀性明显改善.



(a) 仅极小化 $\mathscr{E}^{\text{correctedWinslow}}$ (b) 加入均匀性泛函

图 3.4 有/无均匀性泛函项的参数化结果对比

Fig. 3.4 Comparison of Parameterization Results with/without Uniformity Functional Term

基于以上讨论, 整个参数化流程中所涉及的优化问题都是无约束的, 尤为重要的是避免了计算任务繁重的控制系数, 算法效率大大提高. 为了清晰起见, 我们将本章所提出的构造高质量平面 NURBS 参数化的交替更新方法总结为算法 3.3.

3.4 数值实验与对比分析

在本节中, 我们提供大量实例来展示本章参数化方法的有效性和鲁棒性. 同时, 我们将此方法与现有方法进行比较, 以证明本章方法的优越性.

Algorithm 3.3: 构造高质量平面 NURBS 参数化的交替更新方法

Input: $\boldsymbol{x}(0,\eta), \boldsymbol{x}(1,\eta), \boldsymbol{x}(\xi,0), \boldsymbol{x}(\xi,1)$: 计算域的四条边界 NURBS 曲线;
$\lambda_{\text{unif.}}$: 均匀性泛函的权重;
$dacay_factor$: 参数 δ 的衰减因子;
ε_{rel}: 交替更新控制顶点和权因子的相对误差容忍;
$max_iter_foldElim$: 消除折叠步的最大迭代次数;
$max_iter_alternating$: 交替更新控制顶点和权因子的最大迭代次数.

Output: $\boldsymbol{x}$: 平面 NURBS 参数化.

```
计算区域 Ω 的面积 area(Ω);                                  // (3.18)
构造计算域 Ω 的初始参数化;                                  // 第 3.3.1 节
调用算法 3.2 消除折叠;
for k=0; k ≤ max_iter_alternating; k++ do
    // 式 (3.20)
    固定权因子 ω_i^*, 更新内部控制点 P_i, 并计算 ℰ^pt = min ℰ(x; P_i, ω_i^*);
    // 式 (3.21)
    固定内部控制点 P_ij^*, 更新内部权因子 ω_ij, 并计算 ℰ^wt = min ℰ(x; P_i^*, ω_i);
    if |ℰ^pt − ℰ^wt| < ε_rel then
        break;
    end
end
return 平面 NURBS 参数化 x;
```

3.4.1 算法实现细节

本章所有算法采用 MATLAB® R2020a 和 C++(Visual Studio 2017) 混合编程实现, 计算结果在笔记本电脑 (Intel Core i7-7700HQ CPU, 2.80 GHz, 16GB 内存) 上运行得到. 具体来说, 我们使用 MATLAB 实现整个算法框架, 并用 C++ 计算 NURBS 基函数及其偏导数. 算法3.2和算法3.3中所涉及到的优化问题均使用非线性优化包 NLopt[122] 中的 L-BFGS 方法[123,124] 求解. 对于目标函数中的积分项和线性方程组, 分别使用 Gauss 数值积分和 MATLAB 反除命令计算求解.

尽管通过仔细地调整参数能够尽可能地发挥本章参数化方法的潜力, 不过为了展示其鲁棒性, 我们将算法中的所有参数均设为默认值. 算法 3.1 中的最大分割次数 max_seg 设为 5. 算法 3.2 中参数 δ 的衰减因子 $decay_factor$ 设为 0.1, 消除折叠的最大迭代次数 $max_iter_foldElim$ 设置为 10. 算法 3.3 中均匀性泛函的权重 $\lambda_{\text{unif.}}$ 设置为 1. 交替更新控制顶点和权因子的相对误差容忍 ε_{rel} 和最大迭代次数 $max_iter_alternating$ 分别设置为 0.001 和 5.

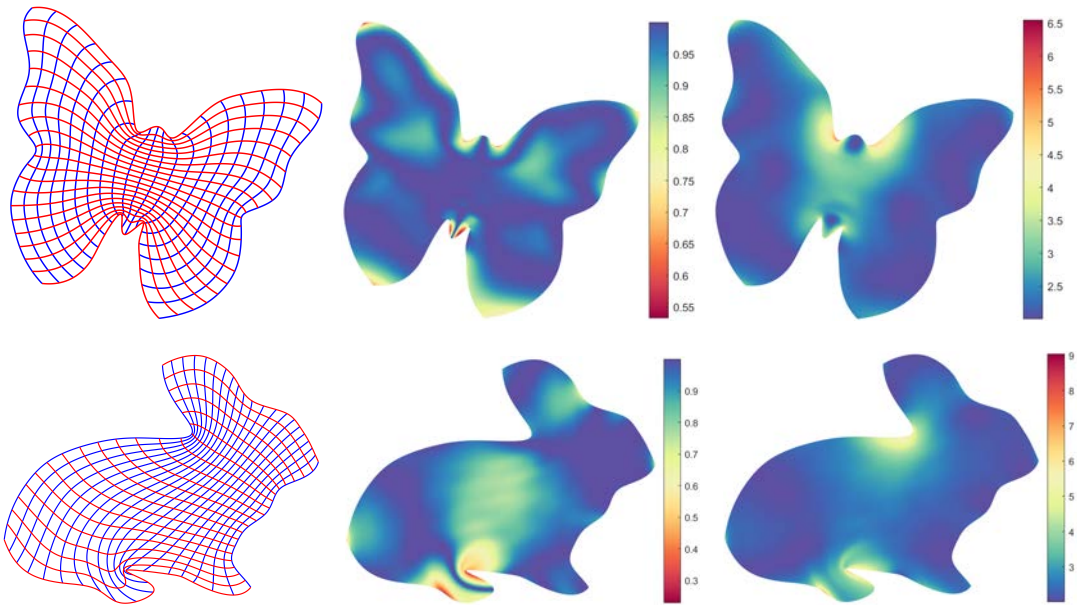
数值试验中所使用的蝴蝶、兔子和海豚模型是通过对文献[1]中所使用的模型修改而来, 而狗和企鹅模型则取自文献[3].

3.4.2 消除折叠的必要性

在本章所涉及到的优化问题中, 目标函数的值由 Gauss 数值积分求得. 如我们在3.3.2节中所提及的, 消除折叠后得到的参数化并不一定在整个计算域上单值. 那么, *这是否意味着算法 3.3 中消除折叠步是不必要的?*

表 3.1 消除折叠的必要性: 不同方法所构造参数化结果的有效性
Tab. 3.1 Necessity of Foldover Elimination: Efficacy of Parameterizations by Different Methods

方法 \ 模型	鸭子	蝴蝶	兔子	飞机	海豚	狗	企鹅
Winslow (3.14)	✗	✗	✗	✗	✗	✗	✗
修正 Winslow (3.15)	✗	✗	✗	✗	✗	✗	✗
Winslow (3.14) + 均匀性 (3.16)	✗	✗	✗	✗	✗	✗	✗
修正 Winslow (3.15) + 均匀性 (3.16)	✗	✗	✗	✗	✗	✗	✗
消除折叠 + Winslow	✓	✓	✓	✓	✗	✗	✗
消除折叠 + 修正 Winslow	✓	✓	✓	✓	✗	✓	✓
消除折叠 + Winslow + 均匀性	✓	✓	✓	✓	✓	✗	✗
消除折叠 + 修正 Winslow + 均匀性	✓	✓	✓	✓	✓	✓	✓



(a) 模型 (b) 比例雅可比 $|\mathcal{J}|_s$ (c) 条件数 $\kappa(\mathcal{J})$

图 3.5 消除折叠 + 修正 Winslow 泛函 + 均匀泛函方法参数化结果及质量指标
Fig. 3.5 Parameterizations and Quality Metrics by Foldover Elimination + Modified Winslow Functional + Uniformity Functional

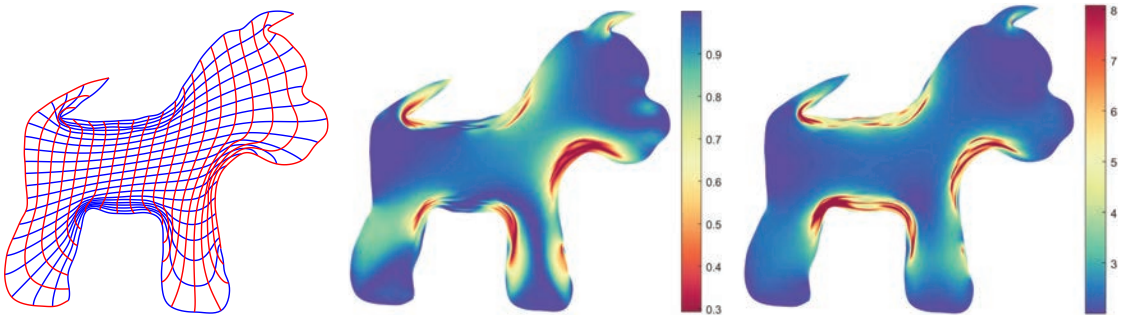
表 3.1 展示了由不同方法所构造的平面 NURBS 参数化结果 $\boldsymbol{x}$ 的有效性. 表中前四行展示了未经消除折叠直接进行参数化质量提升所得到的参数化单值性结果. 这些

方法均无法获得单值的参数化. 在加入消除折叠步之后, 使用原始 Winslow 泛函 (3.14) 的方法 (表 3.1 中第五行) 在海豚, 狗和企鹅模型中得到非单值结果. 未添加均匀性泛函的方法 (表 3.1 中第六行) 在海豚模型中求解失败. 相比之下, 使用消除折叠 + 修正 Winslow 泛函 + 均匀性泛函的方法 (表 3.1 中最后一行) 在所有模型中均成功构造有效的参数化. 这些结果表明, 消除折叠步对参数化方法的成功与否具有至关重要的影响.

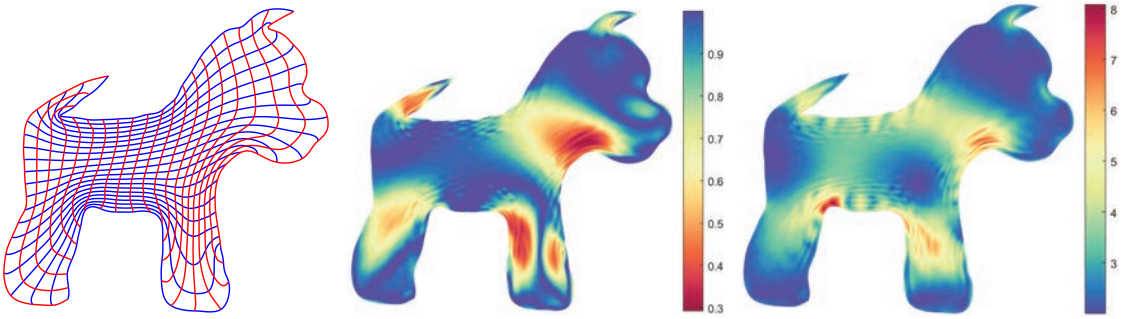
图 3.5 展示了由消除折叠 + 修正 Winslow 泛函 + 均匀性泛函方法所构造蝴蝶和兔子模型的参数化结果及其相应的参数化质量指标: 比例雅可比 $|\mathcal{J}|_s$ 和条件数 $\kappa(\boldsymbol{\mathcal{J}})$. 可以发现, 图中比例雅可比的值均大于零, 这意味着参数化结果是单值的.

3.4.3 权因子的作用

下一个有意思的问题是*更新权因子是否提高了参数化的质量?* 使用 NURBS 可以精确地表示圆锥曲线, 如椭圆、抛物线和球面等. 但除此之外, *改变内部控制顶点所对应的权因子真的有意义吗?*



(a) 固定权因子方法: 参数化结果 (左), 比例雅可比 $|\mathcal{J}|_s$(中) 和相应的条件数 $\kappa(\boldsymbol{\mathcal{J}})$(右)



(b) 交替更新控制顶点和权因子: 参数化结果 (左), 比例雅可比 $|\mathcal{J}|_s$(中) 和相应的条件数 $\kappa(\boldsymbol{\mathcal{J}})$(右)

图 3.6 固定权因子方法与交替更新控制顶点和权因子方法参数化结果对比
Fig. 3.6 Comparison of Parameterizations Obtained by Fixed Weight Factor Method and Alternating Update of Control Points and Weight Factors

为了解答这一问题, 将内部权因子设置为 1, 并在消除折叠之后仅更新内部控制点. 图 3.6 展示了两种方法构造的狗模型参数化结果及其对应的参数化质量指标. 相比之下, 交替更新控制点和权因子的方法生成了比固定权因子方法更优秀的参数化结果.

表 3.2　不同方法所构造参数化结果的量化统计数据
Tab. 3.2　Quantitative Data for Parameterization Results Obtained by Different Methods

模型	p, q, n, m	方法	比例雅可比 $\lvert\mathcal{J}\rvert_s$		条件数 $\kappa(\boldsymbol{\mathcal{J}})$	
			平均值	最小值	平均值	最大值
鸭子	2, 2, 8, 10	非线性约束优化 [53]	0.8819	0.1160	3.1831	32.2954
		变分调和方法 [57]	0.7192	-1.0000	13.1134	$+\infty$
		Teichmüller 映射 [114]	0.8709	-1.0000	5.9432	$+\infty$
		低秩拟共形映射 [1]	0.8850	0.2064	2.9704	16.3875
		固定权因子	0.9212	0.4122	**2.8992**	**6.8609**
		交替更新方法	**0.9214**	**0.4306**	**2.8992**	6.9994
蝴蝶	3, 3, 12, 10	非线性约束优化 [53]	0.9569	0.3280	2.4956	7.7608
		变分调和方法 [57]	0.9137	-1.0000	5.8209	$+\infty$
		Teichmüller 映射 [114]	0.9345	-1.0000	7.1372	$+\infty$
		低秩拟共形映射 [1]	0.9673	0.3206	**2.4184**	14.5630
		固定权因子	**0.9685**	0.5009	2.4682	**6.1219**
		交替更新方法	**0.9685**	**0.5160**	2.4658	6.7475
兔子	3,3,13,10	非线性约束优化 [53]	0.8957	0.1730	2.8256	11.9616
		变分调和方法 [57]	0.8923	-1.0000	5.3268	$+\infty$
		Teichmüller 映射 [114]	0.9100	-1.0000	7.5895	$+\infty$
		低秩拟共形映射 [1]	0.9132	**0.3302**	2.6230	10.1826
		固定权因子	0.9152	-0.0090	2.5671	374.6158
		交替更新方法	**0.9159**	0.2968	**2.5584**	**6.7861**
飞机	2,2,9,8	非线性约束优化 [53]	0.8554	0.3360	4.4755	14.8167
		变分调和方法 [57]	0.6443	-1.0000	18.8121	$+\infty$
		Teichmüller 映射 [114]	0.6879	-1.0000	66.4200	$+\infty$
		低秩拟共形映射 [1]	0.7249	0.2070	4.7270	15.2799
		固定权因子	0.9123	**0.4380**	4.1924	**8.5744**
		交替更新方法	**0.9141**	0.4352	**4.1890**	8.7820
海豚	2,2,12,9	非线性约束优化 [53]	0.5966	0.0823	5.9899	29.2289
		变分调和方法 [57]	0.4866	-0.6861	78.3844	$+\infty$
		Teichmüller 映射 [114]	0.6561	-0.9523	16.3846	$+\infty$
		低秩拟共形映射 [1]	0.5961	0.1647	5.8115	17.7545
		固定权因子	0.7736	**0.2352**	5.0117	11.4117
		交替更新方法	**0.7744**	0.2323	**5.0041**	**11.0161**
狗	3,3,30,30	变分调和方法 [57]	0.4037	-1.0000	21.2539	$+\infty$
		低秩拟共形映射 [1]	0.8256	-0.9995	3.4932	$+\infty$
		固定权因子	0.8331	0.1720	3.6472	17.7726
		交替更新方法	**0.8488**	**0.2930**	**3.2310**	**8.0818**
企鹅	3,3,30,30	变分调和方法 [57]	0.5091	-1.0000	11.9690	$+\infty$
		低秩拟共形映射 [1]	0.8252	-1.0000	4.9784	$+\infty$
		固定权因子	**0.8491**	0.1430	**2.7048**	14.6129
		交替更新方法	**0.8491**	**0.1515**	2.7050	**13.2554**

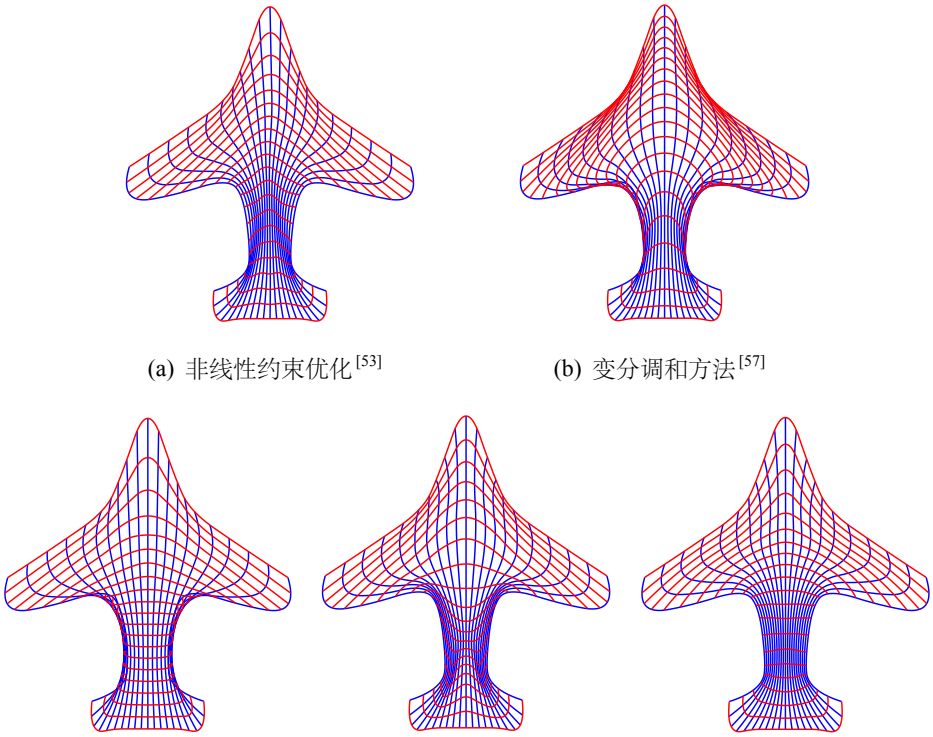
表 3.2 列出了这两种方法所得到的参数化质量指标. 结果表明, 在几乎所有模型

中, 引入权因子更新都能生成更高质量的参数化结果. 在兔子模型中, 固定内部权因子甚至无法得到有效的参数化. 因此, 我们可以得出结论, 交替更新控制点和权因子确实对提高参数化质量产生了积极的影响.

3.4.4 与现有方法的对比分析

在本节中, 我们将本章提出的方法与现有的参数化方法进行比较, 包括非线性约束优化方法[53]、变分调和方法[57]、Teichmüller 映射方法[114] 以及低秩拟共形方法[1].

这几种方法所构造的飞机和海豚模型的参数化结果分别如图 3.7 和图 3.8 所示. 表 3.2 列出了所有方法生成的参数化结果的质量指标. 可以看到, 变分调和方法[57] 和 Teichmüller 映射方法[114] 在本章所展示的所有例子中都不能得到单值的参数化. 低秩拟共形映射方法[1] 在前四个模型中构造了单值的参数化, 而在几何更为复杂的狗和企鹅模型中求解失败. 非线性约束优化方法[53] 在前四个模型中展现了良好的鲁棒性, 但由于无法在合理的时间内生成参数化结果, 后两个模型的单值性无从得知. 相较之下, 我们的方法在所有模型中都能够构造有效的参数化, 具有更优的鲁棒性.



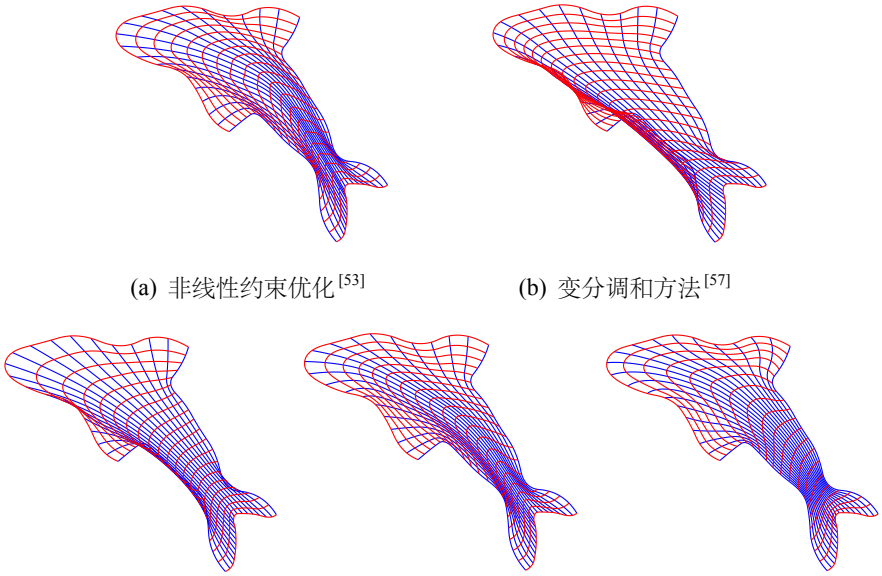
(a) 非线性约束优化[53]　(b) 变分调和方法[57]

(c) Teichmüller 映射[114]　(d) 低秩拟共形映射[1]　(e) 交替更新方法

图 3.7　飞机: 不同参数化方法得到的参数化结果
Fig. 3.7　Plane: Parameterization Results from Different Methods

除单值性外, 参数化质量也是评估各种方法性能的另一个关键标准. 如前所述, 变分调和方法[57] 和 Teichmüller 映射方法[114] 在许多模型中都无法生成单值参数化结果.

因此, 我们仅比较本文方法与非线性约束优化方法[53] 和低秩拟共形方法[1] 的参数化质量. 图 3.9 和图 3.10 分别展示了通过三种方法得到的飞机和海豚模型的参数化质量指标. 两行分别展示了比例雅可比 $|\mathcal{J}|_s$ 和雅可比矩阵的条件数 $\kappa(\mathcal{J})$.

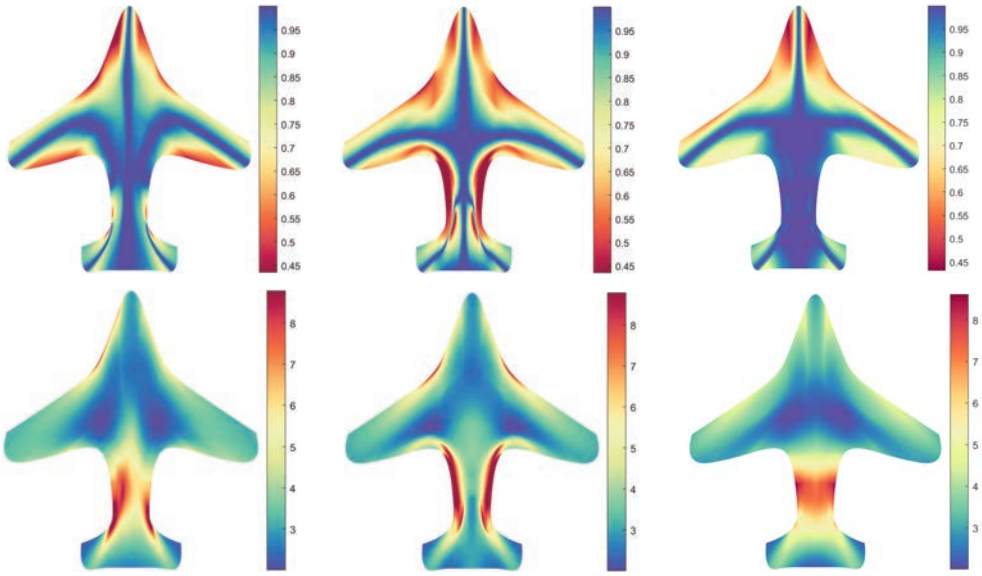


(a) 非线性约束优化[53]　(b) 变分调和方法[57]

(c) Teichmüller 映射[114]　(d) 低秩拟共形映射[1]　(e) 交替更新方法

图 3.8　海豚: 不同参数化方法得到的参数化结果

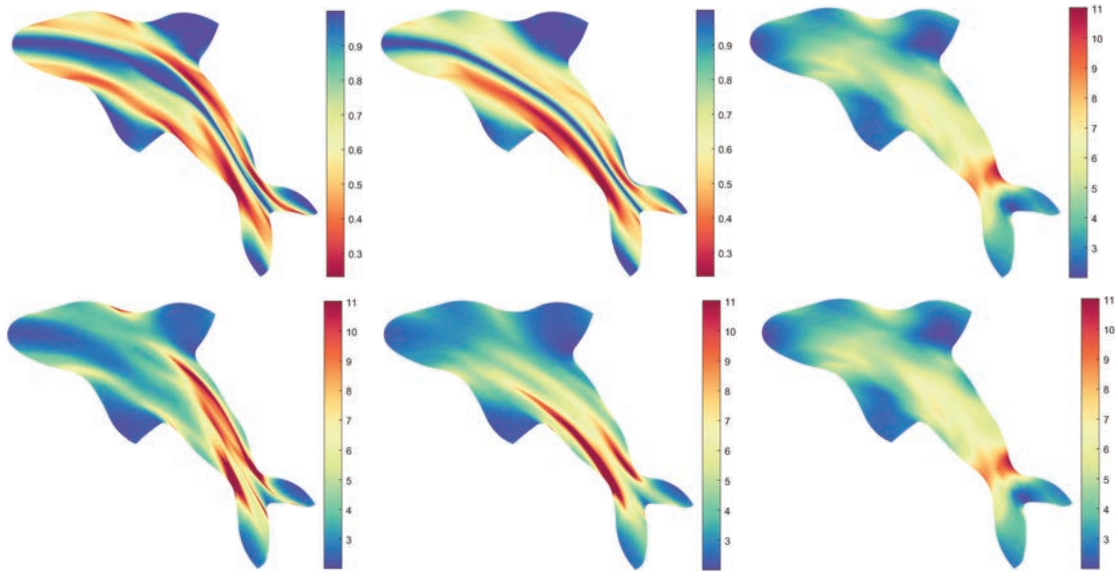
Fig. 3.8　Dolphin: Parameterization Results from Different Methods



(a) 非线性约束优化[53]　(b) 低秩拟共形映射[1]　(c) 交替更新方法

图 3.9　飞机: 比例雅可比 $|\mathcal{J}|_s$(第一行) 和条件数 $\kappa(\mathcal{J})$(第二行)

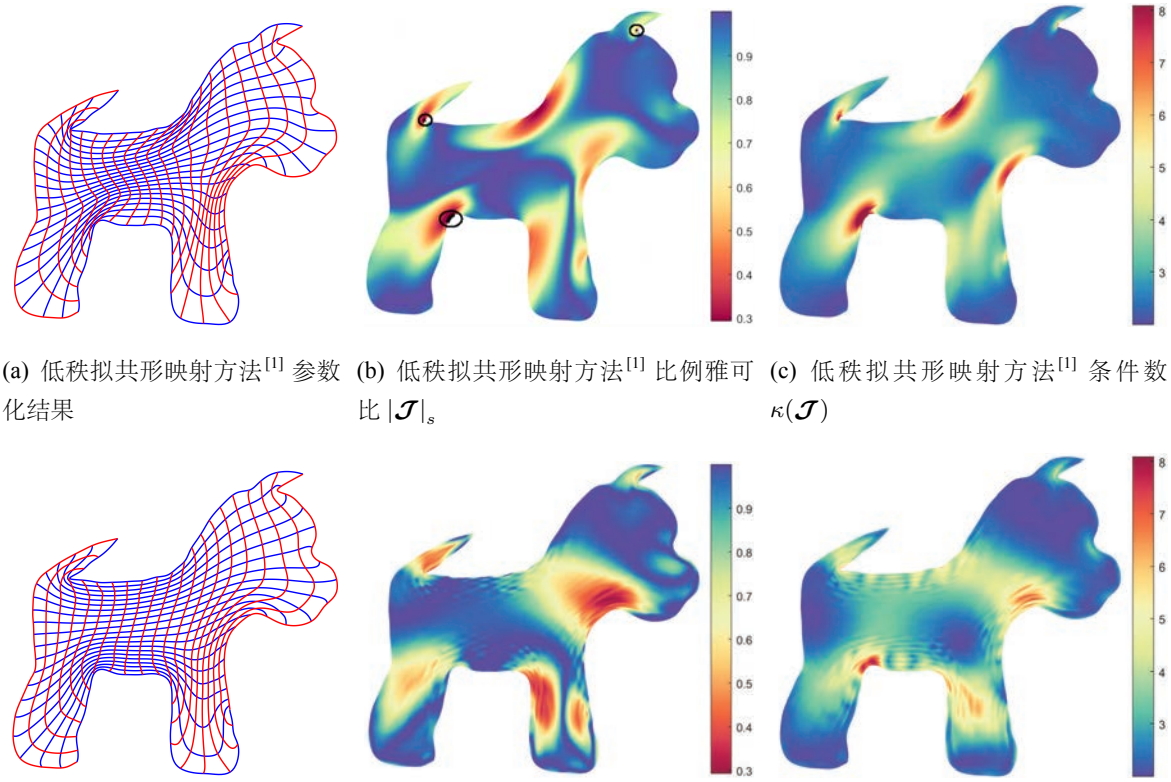
Fig. 3.9　Plane: Scaled Jacobian $|\mathcal{J}|_s$ (First Row) and Condition Number $\kappa(\mathcal{J})$ (Second Row)



(a) 非线性约束优化[53]　(b) 低秩拟共形映射[1]　(c) 交替更新方法

图 3.10　海豚: 比例雅可比 $|\mathcal{J}|_s$(第一行) 和条件数 $\kappa(\mathcal{J})$(第二行)

Fig. 3.10　Dolphin: Scaled Jacobian $|\mathcal{J}|_s$ (First Row) and Condition Number $\kappa(\mathcal{J})$ (Second Row)



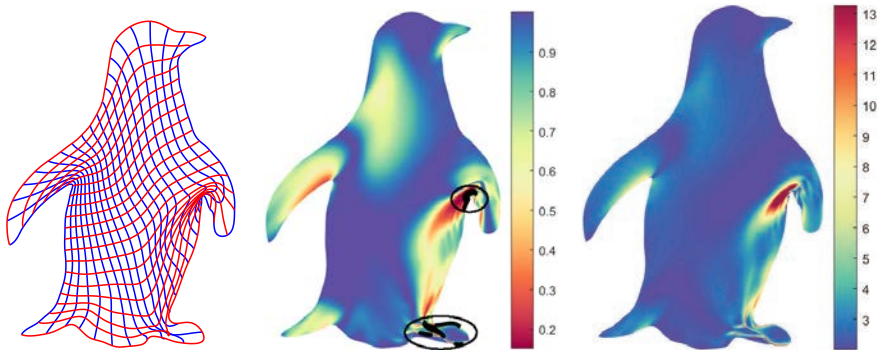
(a) 低秩拟共形映射方法[1] 参数化结果　(b) 低秩拟共形映射方法[1] 比例雅可比 $|\mathcal{J}|_s$　(c) 低秩拟共形映射方法[1] 条件数 $\kappa(\mathcal{J})$

(d) 本章方法参数化结果　(e) 本章方法比例雅可比 $|\mathcal{J}|_s$　(f) 本章方法条件数 $\kappa(\mathcal{J})$

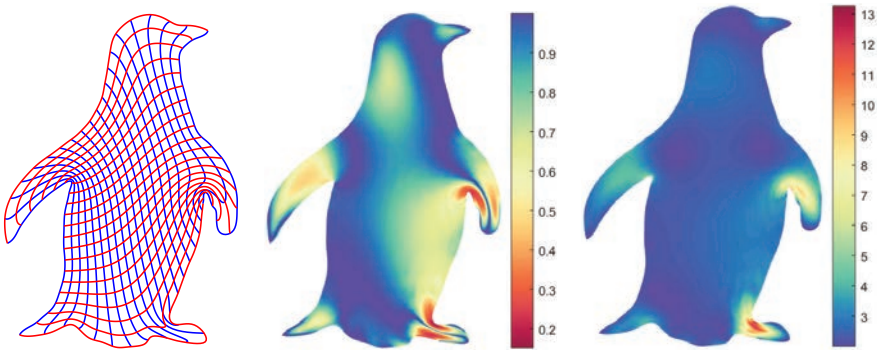
图 3.11　狗: 低秩拟共形映射方法[1] 和本章方法生成的参数化结果与质量比较

Fig. 3.11　Dog: Parameterization Results and Quality Comparison between Low-Rank Quasi-Conformal Mapping Method[1] and Our Method

在前四个模型中, 低秩拟共形映射方法[1] 是本章方法最强的竞争对手. 图 3.11和图 3.12分别展示了低秩拟共形映射方法[1] 在狗和企鹅模型上的表现. 可以看到, 该方法在边界曲率变化较大的区域比例雅可比的值为负, 这意味着得到的参数化结果是非单值的. 而本章方法总是可以获得令人满意的高质量参数化结果.



(a) 低秩拟共形映射方法[1] 参数化结果 (b) 低秩拟共形映射方法[1] 比例雅可比 $|\mathcal{J}|_s$ (c) 低秩拟共形映射方法[1] 条件数 $\kappa(\mathcal{J})$



(d) 本章方法参数化结果 (e) 本章方法比例雅可比 $|\mathcal{J}|_s$ (f) 本章方法条件数 $\kappa(\mathcal{J})$

图 3.12 企鹅: 低秩拟共形映射方法[1] 和本章方法生成的参数化结果与质量比较
Fig. 3.12 Penguin: Parameterization Results and Quality Comparison between Low-Rank Quasi-Conformal Mapping Method[1] and Our Method

表 3.2中展示了更全面的统计数据, 其中用**粗体**标注的数据表示该方法在相应指标上表现最佳. 在对多个模型的分析中, 由本章提出的方法计算得出的比例雅可比 $|\mathcal{J}|_s$ 通常优于其他比较方法, 这表明我们的参数化方法在保持正交性方面表现出色. 不仅如此, 在雅可比矩阵条件数 $\kappa(\mathcal{J})$ 的对比中, 本方法通常也能实现更小的值, 这意味着我们的参数化结果在降低几何扭曲方面具有优势.

3.4.5 计算效率对比

对于实际应用来说, 算法的计算效率也是一个重要指标. 在本节中, 我们将对比不同方法的计算效率.

这里我们不再考虑那些在上述示例中未能获得有效参数化的方法的效率, 而只列出生成有效参数化的示例的运行时间, 如表 3.3 所示. 由于非线性约束优化方法[53] 的每次迭代都需要更新大量的系数约束, 它在运行时间方面表现最差. 然而, 我们的方法相对于低秩拟共形方法展示出更好的效率. 这是因为低秩拟共形映射方法[1] 通过 ADMM 算法迭代求解一个二次优化问题和一个复杂的矩阵补全问题, 耗费了大量时间. 尽管优化权值增加了我们方法的运行时间, 但我们的方法效率仍然是最高的. 此外值得注意的是, 这里的实验数据是采用的是一个串行的 MATLAB 实现得到, 如果完全采用 C/C++ 实现, 则可以进一步提高本文方法的计算效率.

表 3.3　不同参数化方法的计算时间 (秒)
Tab. 3.3　Computation Time (in seconds) for Different Parameterization Methods

方法 \ 模型	鸭子	蝴蝶	兔子	飞机	海豚	狗	企鹅
非线性约束优化[53]	220.27	3567.72	4281.32	304.47	1092.63	-	-
低秩拟共形映射[1]	1.43	2.99	3.06	1.44	1.77	-	-
交替更新方法	**0.41**	**0.87**	**1.20**	**0.43**	**0.85**	**101.16**	**45.75**

3.5　本章小结

本章针对一般平面区域上 NURBS 参数化的单值性提出了一系列充分条件和一个必要条件, 这些条件同时考虑了控制顶点和权因子. 利用所提出的单值性条件, 我们开发了一种实用的单值性检验算法. 通过交替更新内部的控制顶点和权因子, 我们提出了一个高效且鲁棒的高质量平面 NURBS 参数化的障碍函数方法. 数值试验结果表明, 本章的参数化方法具有很好的鲁棒性和计算效率, 并且与现有的方法相比, 能够构造出更高质量的参数化结果.

4 基于罚函数的高质量体参数化方法

4.1 引言

与平面情形相比, 构造适合分析的体参数化更具挑战性. 这一问题的难度主要有三个方面. 第一, 三维计算域的几何复杂性通常远高于平面情况. 第二, 许多在平面情况下表现良好的方法在三维情况下会因为缺乏完善的数学理论支持而失效, 如 Teichmüller 映射方法[114], 拟共形映射方法[1] 和基于调和映射的方法[57,72]. 第三, 体参数化在计算方面的开销要远大于平面参数化. 由于维数的增加, 涉及到的计算量也随之大幅增加，这使得算法效率成为一个关键的考量因素.

在本章中, 我们关注单块 NURBS 体参数化, 如图 4.1 所示. 对于具有高亏格 (high-genus) 的计算域, 多块参数化技术[80,83] 被广泛应用. 这种方法通常首先会把整个计算域分割成多个 NURBS 体块, 然后对每个子块进行参数化. 从这个角度来看, 多块体参数化的质量和计算效率在很大程度上是依赖于单块参数化方法的性能的. 换句话说, 单块参数化方法的优劣将直接影响到高亏格体参数化的整体质量和效率. 因此, 研究并优化单块 NURBS 体参数化方法具有重要的理论和实践意义.

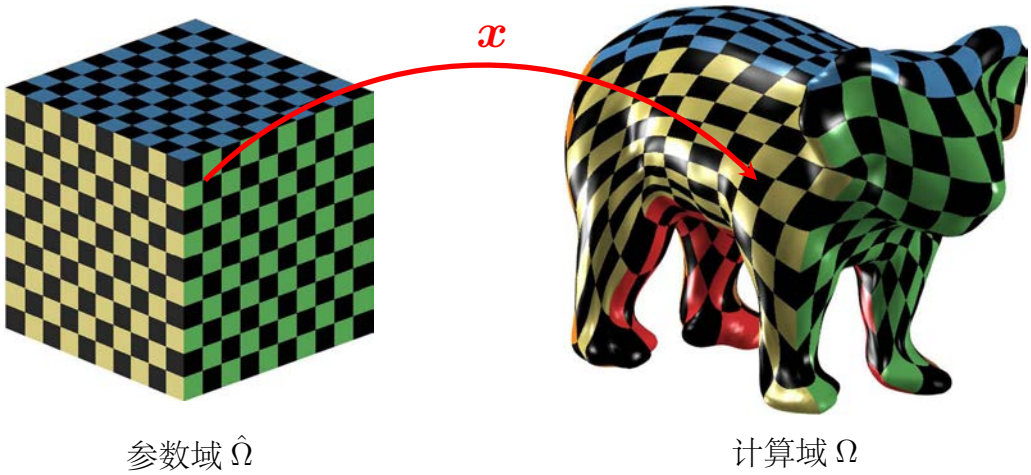


图 4.1 体参数化问题: 构造从参数域 $\hat{\Omega}$ 到计算域 Ω 的单值样条映射 $\boldsymbol{x}$
Fig. 4.1 Volumetric Parameterization Problem: Constructing a Bijective Spline-based Mapping $\boldsymbol{x}$ from the Parameter Domain $\hat{\Omega}$ to the Computational Domain Ω

大体上来说, 根据处理单值性约束的策略, 现有的参数化方法主要可归类为三大类型: 约束优化方法[54,80]、样条拟合方法[125,126] 和障碍函数方法[2,60]. 约束优化方法因需要处理大量的约束条件, 在计算效率上表现并不理想, 特别是在处理三维问题时更加明显. 而样条拟合方法需要首先在计算域上生成一个离散网格, 这实际上违背了等几何分析的初衷. 想较而言, 我们在前一章所介绍的障碍函数方法似乎是最有效的.

然而, 障碍函数方法的一个主要缺点是需要一个已经具备双射性质的初始参数化. 这通常是难以高效实现的, 因此还需要额外进行折叠消除步骤[127]. 更为关键的是, 这一消除折叠步对参数化质量提升的贡献相对有限, 可能会导致不必要的计算开销. 为了解决这些问题, 本章提出一种基于惩罚函数的体参数化方法. 这个方法不仅易于实现, 而且能够有效避免额外的消除折叠步骤. 据作者所知, 这一想法最早是由 Garanzha 在解决网格解扭问题时提出的[66,67]. 最近, 王旭和马维银将这一想法应用到平面参数化问题上[68]. 本章的主要贡献如下:

- 利用散度定理, 给出仅由模型边界表示来计算计算域体积的公式. 计算域的体积 $vol(\Omega)$ 随后被用于构造刻画参数化 $\boldsymbol{x}$ 体积扭曲的目标泛函项;
- 提出了一种新的惩罚函数. 与原有的惩罚函数[66] 相比, 本章所提出的惩罚函数可以有效地降低误差. 结合雅可比正则化思想, 可以直接接受初始的有自相交的参数化而无需额外的消除折叠步骤. 为了提高在基于梯度的优化过程中的计算效率和数值稳定性, 推导出了目标函数的完全解析梯度公式;
- 为了进一步提高方法的计算效率, 我们提议使用一种减缩的数值积分策略. 基于这一策略, 我们提出了一种高效且鲁棒的 NURBS 参数化算法.

4.2 罚函数与雅可比正则化技术

在本章中, 我们主要研究 $\mathbb{R}^3$ 中单连通区域 Ω 的 NURBS 体参数化问题. 给定一个由六张 NURBS 曲面构成的边界表示, 如图 4.1 所示, 包括左侧曲面 $\boldsymbol{x}(0,\eta,\zeta)$, 右侧曲面 $\boldsymbol{x}(1,\eta,\zeta)$, 前侧曲面 $\boldsymbol{x}(\xi,0,\zeta)$, 后侧曲面 $\boldsymbol{x}(\xi,1,\zeta)$, 底部曲面 $\boldsymbol{x}(\xi,\eta,0)$ 以及顶部曲面 $\boldsymbol{x}(\xi,\eta,1)$, 我们的目标是构建一个从参数域 $\hat{\Omega}=[0,1]^3$ 到计算域 Ω 的双射 NURBS 参数化 $\boldsymbol{x}$, 同时尽量减少角度和体积扭曲.

考虑到三维问题中计算量显著增加, 本章仅将内部控制顶点作为优化变量, 以提高计算效率. 为了叙述上的方便, 我们将体参数化中的求和分为两部分:

$$\boldsymbol{x}(\boldsymbol{\xi})=\mathbf{R}^{\mathrm{T}}\mathbf{P}=\underbrace{\sum_{i\in\mathcal{I}_I}\mathbf{P}_iR_i(\boldsymbol{\xi})}_{\text{未知}}+\underbrace{\sum_{j\in\mathcal{I}_B}\mathbf{P}_jR_j(\boldsymbol{\xi})}_{\text{已知}}, \tag{4.1}$$

其中 $\mathbf{P}_i$ 是未知的内部控制点, $\mathbf{P}_j$ 是给定的边界控制点, $\mathcal{I}_I$ 和 $\mathcal{I}_B$ 分别表示内部和边界上控制顶点的索引集, $R_i(\boldsymbol{\xi})$ 是 NURBS 基函数.

因此, 体参数化问题可以归结为: 基于给定的 NURBS 体参数化 $\boldsymbol{x}$ 的边界表示, 构造内部控制顶点 $\mathbf{P}_i$, 使得得到的 $\boldsymbol{x}$ 是双射的, 并且具有尽可能小的角度和体积扭曲.

4.2.1 初始参数化构造

在基于梯度的优化方法中, 一个良好的初始猜测对于数值方法的收敛速度至关重要. 对于参数化问题, 一般采用计算高效的代数参数化方法来构造初始参数化. 第一类是显式方法, 如离散 Coons 方法[51], 它是一种超限插值方法. 第二类是需要求解线性方程组的线性方法, 如弹簧模型[52] 和前向 Laplacian 方法[2].

本章利用前向 Laplacian 方法来构建内部控制点的初始猜测, 即极小化光滑能量

$$\iiint\limits_{\hat{\Omega}} \|\Delta_{\boldsymbol{\xi}} \boldsymbol{x}\|^2 \ \mathrm{d}\hat{\Omega}, \tag{4.2}$$

其中 $\Delta_{\boldsymbol{\xi}} = \frac{\partial^2}{\partial \xi^2} + \frac{\partial^2}{\partial \eta^2} + \frac{\partial^2}{\partial \zeta^2}$ 表示拉普拉斯算子.

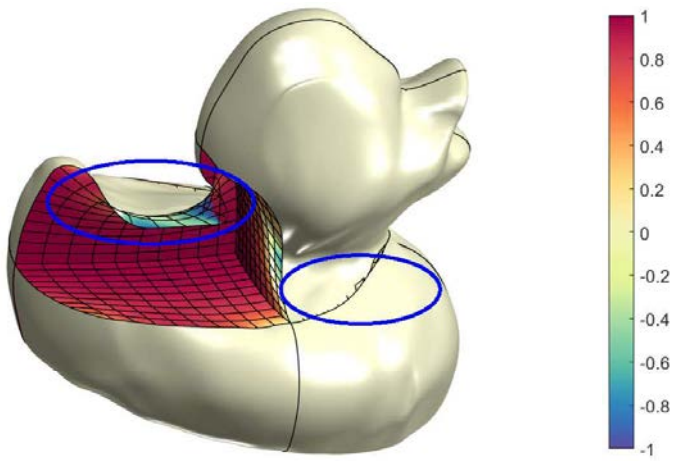


图 4.2 前向 Laplacian 方法得到的初始参数化
Fig. 4.2 Initial Parameterization Obtained by Forward Laplacian Method

二次能量函数 (4.2) 的极小值可以通过求解一个稀疏对称的线性方程组得到, 由此得到的初始参数化如图 4.2 所示. 可以看到, 在小鸭子的背部和腹部存在许多自相交. 下面, 我们将从这样有自相交的初始参数化出发, 构造高质量适合分析的体参数化.

4.2.2 加权目标泛函

参数化的角度扭曲可以由如下的最等距参数化 (Most Isometric ParametrizationS, MIPS) 能量[63]

$$\begin{aligned} \mathscr{E}^{\text{mips}} &= \frac{1}{8}\left(\frac{\sigma_1}{\sigma_2} + \frac{\sigma_2}{\sigma_1}\right)\left(\frac{\sigma_2}{\sigma_3} + \frac{\sigma_3}{\sigma_2}\right)\left(\frac{\sigma_1}{\sigma_3} + \frac{\sigma_3}{\sigma_1}\right) \\ &= \frac{1}{8}\left(\kappa_F^2(\boldsymbol{\mathcal{J}}) - 1\right) \end{aligned} \tag{4.3}$$

所刻画, 其中 $\sigma_1, \sigma_2, \sigma_3$ 表示参数化雅可比矩阵 $\boldsymbol{\mathcal{J}}$ 的奇异值, $\kappa_F^2(\boldsymbol{\mathcal{J}})$ 表示 $\boldsymbol{\mathcal{J}}$ 的 Frobenius 条件数. MIPS 能量描述了参数化的角度扭曲, 但它并没有要求尽可能降低体积扭曲.

映射 $\boldsymbol{x}$ 在某一参数点 $\boldsymbol{\xi}_0$ 上的雅可比行列式 $|\boldsymbol{\mathcal{J}}(\boldsymbol{\xi}_0)|$ 刻画了其在该点附近引起的体积的变化量. 在最理想的情况下, 雅可比行列式 $|\boldsymbol{\mathcal{J}}|$ 在整个参数域 $\hat{\Omega}$ 上的值都是相同的, 这意味着此时不存在体积扭曲. 由于参数域 $\hat{\Omega} = [0,1]^3$, 我们定义如下的均匀能量泛函来描述体积扭曲:

$$\mathscr{E}^{\text{vol}} = \frac{vol(\Omega)}{|\boldsymbol{\mathcal{J}}|} + \frac{|\boldsymbol{\mathcal{J}}|}{vol(\Omega)}, \tag{4.4}$$

其中 $vol(\Omega)$ 表示计算域 Ω 的体积.

此时, 一个新的问题出现了, *如何计算* $vol(\Omega)$? 一个简单的方法如下:

$$vol(\Omega) = \iiint\limits_{\Omega} 1 \ \mathrm{d}\Omega = \iiint\limits_{\hat{\Omega}} |\boldsymbol{\mathcal{J}}| \ \mathrm{d}\hat{\Omega}. \tag{4.5}$$

不幸的是, 上述计算公式 (4.5) 需要体参数化的内部参数信息, 这在给定的 CAD 模型边界表示中是未知的.

为了解决这一问题, 我们考虑如何仅使用给定的边界表示来计算体积 $vol(\Omega)$. 应用散度定理, 我们得到:

$$\begin{aligned}
vol(\Omega) &= \frac{1}{3} \iint\limits_{\partial\Omega} (x_1 \ \mathrm{d}x_2\mathrm{d}x_3 + x_2 \ \mathrm{d}x_3\mathrm{d}x_1 + x_3 \ \mathrm{d}x_1\mathrm{d}x_2) \\
&= \frac{1}{3} \iint\limits_{\partial\hat{\Omega}} \begin{vmatrix} x_1 & x_2 & x_3 \\ x_{1,u} & x_{2,u} & x_{3,u} \\ x_{1,v} & x_{2,v} & x_{3,v} \end{vmatrix} \mathrm{d}u\mathrm{d}v \\
&= \frac{1}{3} \int_0^1 \int_0^1 [\Psi_{\eta,\zeta}(1,\eta,\zeta) - \Psi_{\eta,\zeta}(0,\eta,\zeta)]\mathrm{d}\eta\mathrm{d}\zeta \\
&\quad + \frac{1}{3} \int_0^1 \int_0^1 [\Psi_{\xi,\zeta}(\xi,0,\zeta) - \Psi_{\xi,\zeta}(\xi,1,\zeta)]\mathrm{d}\xi\mathrm{d}\zeta \\
&\quad + \frac{1}{3} \int_0^1 \int_0^1 [\Psi_{\xi,\eta}(\xi,\eta,1) - \Psi_{\xi,\eta}(\xi,\eta,0)]\mathrm{d}\eta\mathrm{d}\zeta,
\end{aligned} \tag{4.6}$$

其中

$$\Psi_{u,v}(\xi,\eta,\zeta) = \begin{vmatrix} x_1(\xi,\eta,\zeta) & x_2(\xi,\eta,\zeta) & x_3(\xi,\eta,\zeta) \\ x_{1,u}(\xi,\eta,\zeta) & x_{2,u}(\xi,\eta,\zeta) & x_{3,u}(\xi,\eta,\zeta) \\ x_{1,v}(\xi,\eta,\zeta) & x_{2,v}(\xi,\eta,\zeta) & x_{3,v}(\xi,\eta,\zeta) \end{vmatrix}.$$

请注意, 式 (4.6) 中仅使用了给定的边界表示, 而无需额外信息. 与式 (4.5) 相比, 其计算降低了一个维度, 因此大大降低了计算成本.

最终, 目标泛函可以表示为 MIPS 能量和均匀能量泛函 (4.4) 的加权和在参数域 $\hat{\Omega}$ 上的积分:

$$\mathscr{E} = \iiint\limits_{\hat{\Omega}} \left(\lambda_{\text{mips}}\mathscr{E}^{\text{mips}} + \lambda_{\text{vol}}\mathscr{E}^{\text{vol}}\right) \mathrm{d}\hat{\Omega}, \tag{4.7}$$

其中, λ_{mips} 和 λ_{vol} 是用户指定的参数, 用于平衡体参数化 $\boldsymbol{x}$ 的角度和体积扭曲.

4.2.3 雅可比正则化技术

回想在 MIPS 能量 (4.3) 中所出现的 Frobenius 条件数, 我们有

$$\begin{aligned}\kappa_F^2(\boldsymbol{\mathcal{J}}) &= \|\boldsymbol{\mathcal{J}}\|_F^2\|\boldsymbol{\mathcal{J}}^{-1}\|_F^2 \\ &= (\sigma_1^2+\sigma_2^2+\sigma_3^2)(\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}+\frac{1}{\sigma_3^2}) \\ &= \frac{(\sigma_1^2+\sigma_2^2+\sigma_3^2)\left(\sigma_2^2\sigma_3^2+\sigma_1^2\sigma_3^2+\sigma_1^2\sigma_2^2\right)}{|\boldsymbol{\mathcal{J}}|^2}.\end{aligned} \tag{4.8}$$

可以观察到, 雅可比行列式 $|\boldsymbol{\mathcal{J}}|$ 出现在分母的位置上. 同时, 它也出现在均匀能量泛函 (4.4) 的分母中. 当 $|\boldsymbol{\mathcal{J}}|$ 接近零时, 积分项趋向于无穷大. 正如我们在第 3 章所述, 目标泛函的这一特性形成了一个屏障, 从而能够有效地抑制参数化的自相交. 然而, 前提条件是需要找到一个已经是双射的初始参数化. 为此, 文献 [2] 和 [127] 增加了一个消除折叠的步骤来解决这一问题.

然而, 对于复杂的计算域, 这种双射参数化通常难以高效地获得. 为了接受无效的初始参数化, 并在优化过程中消除折叠, 本文提出了一种新的惩罚函数:

$$\chi(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta) = \begin{cases} \varepsilon\cdot e^{\beta(|\boldsymbol{\mathcal{J}}|-\varepsilon)} & \text{若 } |\boldsymbol{\mathcal{J}}| \le \varepsilon, \\ |\boldsymbol{\mathcal{J}}| & \text{若 } |\boldsymbol{\mathcal{J}}| > \varepsilon, \end{cases} \tag{4.9}$$

其中 ε 是一个小的正数, β 是用来控制惩罚函数坡度的惩罚因子.

如图 4.3(a) 所示, 如果 $|\boldsymbol{\mathcal{J}}| < \varepsilon$, 则 $\chi(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta)$ 等于一个较小的正数; 如果 $|\boldsymbol{\mathcal{J}}| \ge \varepsilon$, 则严格等于雅可比行列式 $|\boldsymbol{\mathcal{J}}|$. 因此, 直观上来讲, $\frac{1}{\chi^2(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta)}$ 具有很大的值来惩罚负的雅可比行列式, 具有较小的值来接收正的雅可比行列式.

而惩罚函数 $\chi(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta)$ 关于雅可比行列式 $|\boldsymbol{\mathcal{J}}|$ 的导数为

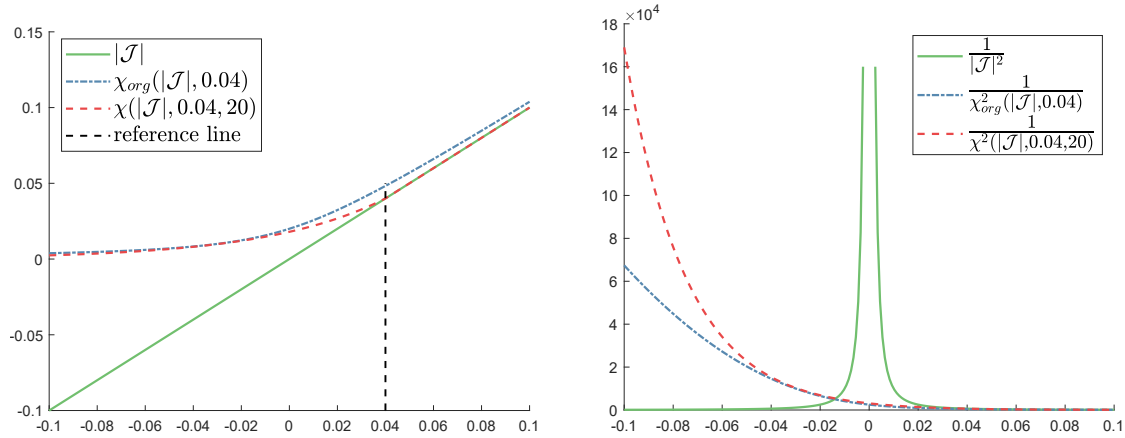
$$\frac{\partial\chi(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta)}{\partial|\boldsymbol{\mathcal{J}}|} = \begin{cases} \beta\cdot\chi(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta) & \text{若 } |\boldsymbol{\mathcal{J}}| \le \varepsilon, \\ 1 & \text{若 } |\boldsymbol{\mathcal{J}}| > \varepsilon. \end{cases} \tag{4.10}$$

可以看到, 其导数要么是它本身的常数倍, 要么是 1, 这使得计算更加高效.

注 4.1. 针对网格解扭问题, Garanzha 提出了一个惩罚函数 χ_{org}[66]:

$$\chi_{org}(|\boldsymbol{\mathcal{J}}|,\varepsilon)=\frac{|\boldsymbol{\mathcal{J}}|+\sqrt{\varepsilon^2+|\boldsymbol{\mathcal{J}}|^2}}{2}, \tag{4.11}$$

其中 ε 是一个正数. 然而, 如图 4.3(a)所示, 即使当 $|\boldsymbol{\mathcal{J}}|>\varepsilon$ 时, 罚函数 χ_{org} 也不严格等于雅可比行列式 $|\boldsymbol{\mathcal{J}}|$. 这会引入额外的误差, 在实际应用中可能并不理想.



(a) 雅可比行列式与不同的惩罚函数　　(b) 雅可比行列式与惩罚函数倒数的平方

图 4.3　惩罚函数与雅可比正则化技术

Fig. 4.3　Penalty Function and Jacobian Regularization Technique

基于雅可比正则化思想, 将 Frobenius 条件数 (4.8) 中分母位置的雅可比行列式替换为惩罚函数 $\chi(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta)$, 我们有:

$$\begin{aligned}\kappa^2_{F,\varepsilon}(\mathcal{J})&=\frac{(\sigma_1^2+\sigma_2^2+\sigma_3^2)(\sigma_2^2\sigma_3^2+\sigma_1^2\sigma_3^2+\sigma_1^2\sigma_2^2)}{\chi^2(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta)}\\&=\kappa^2_F(\mathcal{J})\cdot\frac{|\boldsymbol{\mathcal{J}}|^2}{\chi^2(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta)}.\end{aligned} \tag{4.12}$$

于是, 我们得到如下的修正 MIPS 能量函数

$$\begin{aligned}\mathscr{E}^{\text{mips,c}}&=\frac{1}{8}\left(\kappa^2_{F,\varepsilon}(\boldsymbol{\mathcal{J}})-1\right)\\&=\frac{1}{8}\left(\kappa^2_F(\boldsymbol{\mathcal{J}})\cdot\frac{|\boldsymbol{\mathcal{J}}|^2}{\chi^2(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta)}-1\right).\end{aligned} \tag{4.13}$$

类似地, 我们对均匀能量泛函 (4.4) 也进行修正:

$$\mathscr{E}^{\text{vol,c}}=\frac{vol(\Omega)}{\chi(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta)}+\frac{\chi(|\boldsymbol{\mathcal{J}}|,\varepsilon,\beta)}{vol(\Omega)}. \tag{4.14}$$

最终, 修正的加权目标泛函为

$$\mathscr{E}^c = \iiint\limits_{\hat{\Omega}} \left(\lambda_{\text{mips}}\mathscr{E}^{\text{mips,c}} + \lambda_{\text{vol}}\mathscr{E}^{\text{vol,c}}\right) \, \mathrm{d}\hat{\Omega}. \tag{4.15}$$

4.3 算法效率上的几点考虑

4.3.1 灵敏度分析

对于具有大量优化变量的非线性优化问题, 一阶求解方法被广泛使用, 如拟牛顿方法. 在这类方法中, 目标函数梯度的精度对计算的稳定性和效率有至关重要的影响. 因此, 本节致力于分析和计算修正的加权目标泛函 (4.15) 的梯度.

为简洁起见, 我们用 ∂_p 表示关于变量 p 的偏导数, 并用 χ 简记正则化函数 $\chi(|\boldsymbol{\mathcal{J}}|, \varepsilon, \beta)$. 首先, 我们有以下基本关系式:

$$\partial_p \boldsymbol{\mathcal{J}} = \mathbf{G}\, \partial_p \mathbf{P}. \tag{4.16}$$

根据雅可比公式, 我们可以得到雅可比行列式 $|\boldsymbol{\mathcal{J}}|$ 的偏导数:

$$\partial_p |\boldsymbol{\mathcal{J}}| = |\boldsymbol{\mathcal{J}}| \; trace(\boldsymbol{\mathcal{J}}^{-1}\partial_p \boldsymbol{\mathcal{J}}), \tag{4.17}$$

其中 $trace(\cdot)$ 表示矩阵的迹.

此外, Frobenius 条件数 $\kappa_F^2(\boldsymbol{\mathcal{J}})$ 的偏导数[63] 为:

$$\partial_p \kappa_F^2(\boldsymbol{\mathcal{J}}) = 2\, trace((\|\boldsymbol{\mathcal{J}}^{-1}\|_F^2 \boldsymbol{\mathcal{J}}^{\mathrm{T}} - \|\boldsymbol{\mathcal{J}}\|_F^2 (\boldsymbol{\mathcal{J}}\boldsymbol{\mathcal{J}}^{\mathrm{T}}\boldsymbol{\mathcal{J}})^{-1})\partial_p \boldsymbol{\mathcal{J}}). \tag{4.18}$$

利用微分链式法则, 可以计算出修正的 Frobenius 条件数 (4.12) 的偏导数为

$$\partial_p \kappa_{F,\varepsilon}^2(\boldsymbol{\mathcal{J}}) = \frac{\partial_p \kappa_F^2(\boldsymbol{\mathcal{J}})\, |\boldsymbol{\mathcal{J}}|^2 + 2\kappa_F^2(\boldsymbol{\mathcal{J}})\, |\boldsymbol{\mathcal{J}}|\, \partial_p |\boldsymbol{\mathcal{J}}|}{\chi^2} - 2\kappa_{F,\varepsilon}^2(\boldsymbol{\mathcal{J}}) \frac{\partial \chi}{\partial |\boldsymbol{\mathcal{J}}|} \frac{\partial_p |\boldsymbol{\mathcal{J}}|}{\chi}. \tag{4.19}$$

接下来, 可以通过以下方式计算均匀能量泛函 $\mathscr{E}^{\text{vol,c}}$ 关于变量 p 的偏导数:

$$\partial_p \mathscr{E}^{\text{vol,c}} = \left(\frac{1}{vol(\Omega)} - \frac{vol(\Omega)}{\chi^2}\right) \frac{\partial \chi}{\partial |\boldsymbol{\mathcal{J}}|} \partial_p |\boldsymbol{\mathcal{J}}|. \tag{4.20}$$

根据式 (4.17), (4.19) 和 (4.20), 最终可以得到修正的加权目标泛函 (4.15) 关于变量 p 的偏导数如下:

$$\partial_p \mathscr{E}^c = \iiint\limits_{\hat{\Omega}} \left(\frac{1}{8}\lambda_{\text{mips}} \partial_p \kappa_{F,\varepsilon}^2(\boldsymbol{\mathcal{J}}) + \lambda_{\text{vol}}\, \partial_p \mathscr{E}^{\text{vol,c}}\right) \, \mathrm{d}\hat{\Omega}. \tag{4.21}$$

注 4.2. 从实现的角度来看, 对于那些 $|\boldsymbol{\mathcal{J}}| > \varepsilon$ 的 Gauss 积分点, 使用本章所提出的惩

罚函数 (4.9), 无需修改雅可比矩阵. 这是因为在这些 Gauss 点, χ 等于其雅可比行列式 $|\mathcal{J}|$ 的值. 因此, 本章所提出的惩罚函数 χ 可以进一步降低计算成本.

4.3.2 高效的减缩积分方案

与线性网格单元相比, 样条参数化具有更高的次数, 因此往往需要更多的数值积分点来达到令人满意的数值积分精度. 然而, 这同时给计算效率带来了巨大的挑战. 为了提高计算效率, 我们开发了一种启发式的减缩积分策略.

从图 4.2 上我们可以看出, 雅可比行列式在边界附近变化很大, 但在内部总是变化相对平缓. 目标泛函值也有类似的表现. 利用这一观察, 我们启发式地在边界附近的单元和内部单元采用不同数量的 Gauss 积分点. 具体来说, 对于靠近边界的单元, 我们在每个方向上采用次数 +1 个积分点; 而对于内部的单元, 我们在每个方向上使用 $\max\{2,\ degree-1\}$ 个积分点, 其中 $degree$ 表示该方向上基函数的次数.

为了更直观地展示减缩积分策略, 图 4.4 展示了平面区域上双三次 NURBS 参数化 Gauss 积分点的分布: 在靠近边界的最外层单元中使用 4×4 个 Gauss 积分点, 而在内部单元中使用 2×2 个 Gauss 积分点. 对于三维情形, 我们采取类似的数值积分方案.

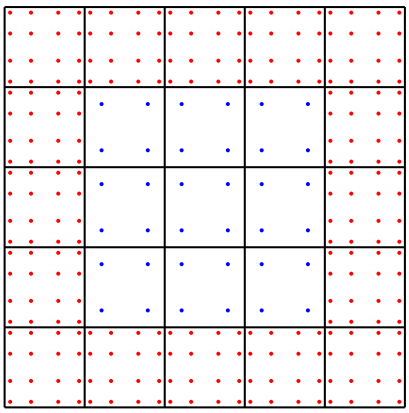


图 4.4 平面区域上双三次 NURBS 参数化减缩积分方案

Fig. 4.4 Reduced Quadrature Scheme for Bicubic NURBS Parameterization on Planar Domain

此外, 由于 Gauss 积分点的位置在每次迭代中没有改变, 我们在迭代之前预先计算 NURBS 基函数在 Gauss 积分点处的值和导数, 以进一步提高计算效率. 我们将在第 4.4.4 节中展示所提出的减缩积分策略与高精度方法之间的比较.

4.3.3 数值优化算法

在构造初始参数化之后, 本章所提出的体参数化方法可以重新表述为以下无约束非线性优化问题:

$$\underset{\mathbf{P}_i,\ i\in\mathcal{I}_I}{\arg\min}\ \mathscr{E}^c = \iiint_{\hat{\Omega}} \left(\lambda_{\text{mips}}\mathcal{E}^{\text{mips,c}} + \lambda_{\text{vol}}\mathcal{E}^{\text{vol,c}}\right)\ \mathrm{d}\hat{\Omega}, \tag{4.22}$$

其中 $\mathbf{P}_i$ 是未知的内部控制顶点.

对于大规模问题, 由于其较低的计算成本和存储需求, 一阶优化方法通常更受欢迎. 在本文中, 我们采用了标准的 L-BFGS 方法[124] 来求解非线性优化问题 (4.22). 此外, 我们使用带有强 Wolfe 准则的回溯线搜索方法[128] 来寻找合适的步长.

在求解中, 我们采用了三个终止准则:

1) 连续两次迭代间目标函数值的相对变化小于给定的阈值 $\delta_{\mathscr{E}^c}$ 时, 即

$$\left|\mathscr{E}^c_{(k-1)} - \mathscr{E}^c_{(k)}\right| < \delta_{\mathscr{E}^c} \left|\mathscr{E}^c_{(k)}\right|, \tag{4.23}$$

其中 $\mathscr{E}^c_{(k)}$ 表示第 k 次迭代目标函数 $\mathscr{E}^c$ 的值, $\delta_{\mathscr{E}^c}$ 是一个用户指定的误差容忍;

2) 当梯度的相对变化量小于给定的阈值 $\delta_{\nabla\mathscr{E}^c}$ 时, 即

$$\left\|\nabla\mathscr{E}^c_{(k)}\right\| < \delta_{\nabla\mathscr{E}^c} \max\left(1,\ \left\|\mathbf{P}_{(k)}\right\|\right), \tag{4.24}$$

其中 $\nabla\mathscr{E}^c_{(k)}$ 表示第 k 次迭代目标函数 $\mathscr{E}^c$ 的梯度, $\mathbf{P}_{(k)}$ 是第 k 次迭代中的控制顶点, $\delta_{\nabla\mathscr{E}^c}$ 是一个用户指定的误差容忍;

3) 超过最大迭代次数 max_iter, 则停止迭代. 通常, 我们将 max_iter 设置为一个较大的值, 以确保算法在最大迭代次数之前收敛.

4.3.4 基于罚函数的参数化算法总览

综合上述分析, 本文提出的体积参数化方法的流程如算法4.1所示. 由于可以接受有自相交的初始参数化, 所提出的方法非常简单且高效.

注 4.3. 需要注意的是, 由于指数函数的存在, 所提出的方法存在数值溢出的风险, 特别是当初始参数化中包含具有较大雅可比行列式绝对值 $|\mathcal{J}|$ 的点时. 尽管在我们的数值实验中从未发生过, 但为了更好的数值稳定性, 这个问题可以通过以下三步策略来解决: 首先, 缩小计算域的包围盒至单位立方体; 执行本章所提出的体参数化技术; 最后, 将计算域的包围盒恢复至原始体积. 同时, 这一策略也有助于实现一致的收敛准则.

4.4 数值实验与对比分析

在本节中, 我们将通过数值实验来评估和对比本章所提出的基于罚函数的高质量体参数化方法. 第 4.4.1 节解释了算法的实现细节. 在接下来的几个小节中, 我们分别考虑了初始参数化对参数化结果的影响 (第 4.4.2 节), 算法的鲁棒性 (第 4.4.3 节), 平衡参数 λ_{mips} 和 λ_{vol} 对参数化结果的影响 (第 4.4.5 节), 减缩积分策略的有效性 (第 4.4.4 节), 以及和现有方法的对比 (第 4.4.6 节).

Algorithm 4.1: 基于罚函数的 NURBS 体参数化方法

Input: $\partial\Omega$: 由六个 NURBS 曲面组成的 CAD 模型的边界表示;
$\lambda_{\text{mips}}, \lambda_{\text{vol}}$: 平衡 MIPS 能量和均匀能量泛函的系数;
$\varepsilon > 0, \beta$: 惩罚函数中的参数;
$\delta_{\nabla\mathscr{E}^c}$: 相对梯度变化容忍;
$\delta_{\mathscr{E}^c}$: 相对目标泛函值变化容忍;
max_iter: 最大迭代次数.

Output: $\boldsymbol{x}$: 体 NURBS 参数化.

```
将内部权因子置为 1;
通过求解问题 (4.2) 对应的线性方程组计算内部控制顶点;  // 第 4.2.1 节
利用区域 Ω 的边界表示计算其体积;                        // (4.6)
预计算 Gauss 权重;                                    // 第 4.3.2 节
预计算 NURBS 基函数在 Gauss 积分点上的值和导数值;      // 第 4.3.2 节
for k=0; k ≤ max_iter; k++ do
    计算修正的加权目标函数 ℰ^c_(k) 的值;                // (4.15)
    计算修正的加权目标函数的梯度 ∇ℰ^c_(k);              // (4.21)
    执行 L-BFGS 迭代以计算下降方向和步长;
    更新体 NURBS 参数化 x 的内部控制顶点;               // 第 4.3.3 节
    // 终止准则测试: (4.23) 或 (4.24)
    if |ℰ^c_(k−1) − ℰ^c_(k)| < δ_ℰc |ℰ^c_(k)| 或 ‖∇ℰ^c_(k)‖ < δ_∇ℰc max(1, ‖P_(k)‖) then
        break;
    end
end
return 体 NURBS 参数化 x;
```

4.4.1 算法实现细节

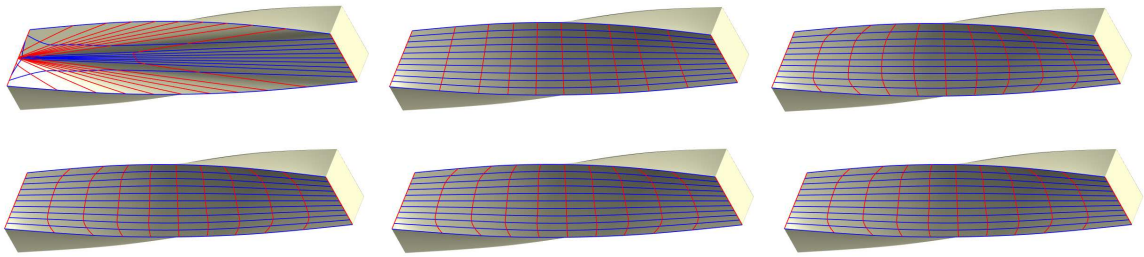
我们使用 MATLAB® R2020b 和 C++ 混合编程实现了所提出的参数化方法, 所有的计算结果在一台配置了英特尔 Core i7-7700HQ CPU, 2.80 GHz 和 16GB 内存的笔记本电脑上运行得到. 具体来说, 整个算法框架使用 MATLAB 实现, 而 NURBS 基函数及其偏导数的计算使用 C++ 实现. 此外, 我们使用了 L-BFGS 方法的一个 MATLAB 接口[129], 其底层由 C 语言实现.

在本章所提出的算法中, 仅需设置很少量的几个参数. 尽管可以通过细致地调整参数来提高参数化质量和收敛速度, 但我们仍将所有相关参数设为默认值. 对于式 (4.9) 中的惩罚函数 χ, 我们分别将参数 ε 和 β 设置为 $0.05\,vol(\Omega)$ 和 1. 同时, 我们将 λ_{mips} 设为 4. 默认情况下, 参数 λ_{vol} 设为 1. 但在最后六个复杂模型中, 将其设置为 10 以显著降低体积扭曲 (适用于从第 4.4.4 节到第 4.4.6 节的实验). 至于算法的终止准则, 我们将 $\delta_{\nabla\mathscr{E}^c}$ 和 $\delta_{\mathscr{E}^c}$ 分别设为 10^{-3}, 最大迭代次数 max_iter 设为 100.

4.4.2 初始参数化对参数化结果的影响

在本小节中, 我们考虑不同初始参数化对参数化结果的影响, 并为如何选择初始化方法提供一定的指导. 具体地, 我们采用以下三种初始参数化构造方法:

- 相同点方法: 将所有的内部控制点放置在相同的位置 $(0,0,0)^{\mathrm{T}}$ 上;
- 离散 Coons 方法: 采用离散 Coons 体方法[49] 构造内部的控制顶点;
- 前向 Laplacian 方法: 如第 4.2.1 节中所述.



(a) 相同点方法 (b) 离散 Coons 方法 (c) 前向 Laplacian 方法

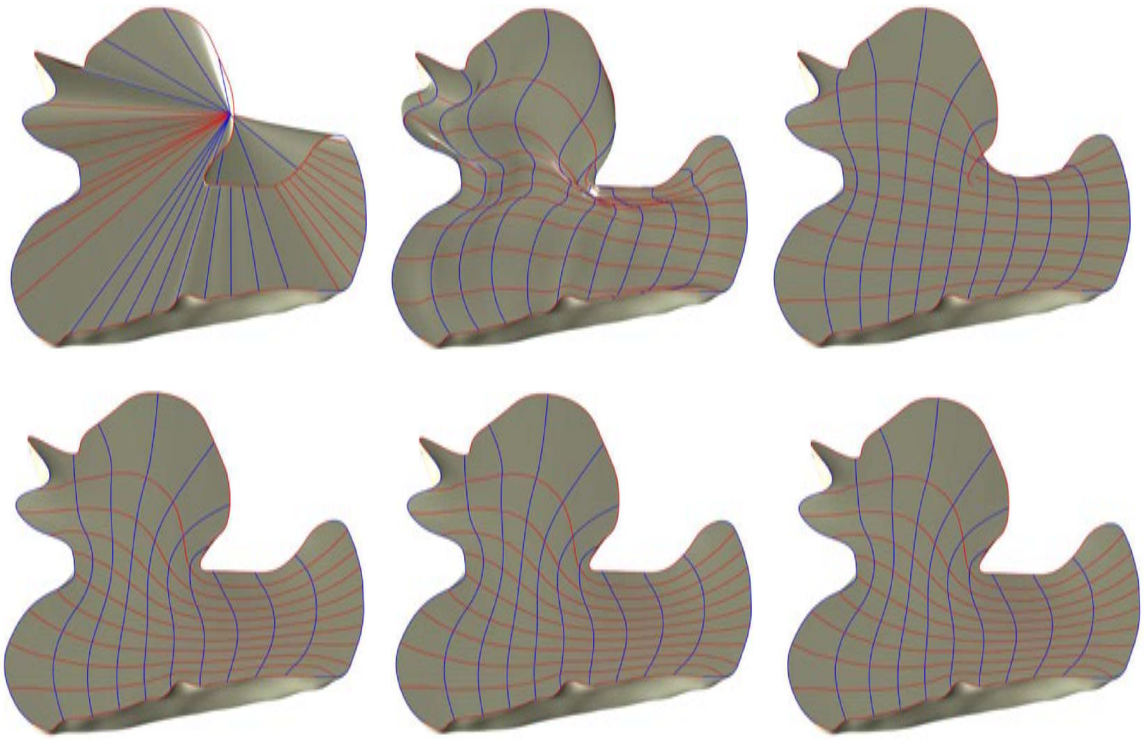
图 4.5 Rot90: 由不同初始参数化得到的参数化结果
Fig. 4.5 Rot90: Parameterization Results from Different Initializations

考虑一个由三次 NURBS 体参数化的长方体, 其中每个参数方向有 6 个控制点, 即 $p=q=r=3$ 且 $n=m=l=5$. 如图 4.5 所示, 我们按照 $k\cdot 90°/5$ $(k=0,1,\ldots,5)$ 从左到右旋转第 k 个平面的控制顶点. 图 4.5 的第一行展示了各种初始化方法构造的初始参数化, 而第二行分别展示了使用所提出方法获得的参数化结果. 更详尽的的统计数据列在表 4.1 中. 尽管采用了不同的初始化方法, 但所得参数化质量相差不大.

表 4.1 不同初始参数化得到的参数化结果的量化统计数据, 时间单位 (秒)
Tab. 4.1 Quantitative Data of Parameterizations from Different Initializations, Time in Seconds

模型	初始化方法	$T_{\text{初始化}}$	迭代数	$T_{\text{总}}$	比例雅可比		均匀性	
					最小值	平均值	最大值	平均值
Rot90	相同点方法	**0.0004**	55	0.7694	0.5827	0.9203	**0.4130**	0.0275
	离散 Coons	0.0017	14	0.2720	0.5878	**0.9206**	**0.4130**	0.0275
	前向 Laplacian	0.0079	**13**	**0.2586**	**0.5913**	0.9203	**0.4130**	**0.0274**
鸭子	相同点方法	**0.0006**	300	41.5663	**0.0028**	0.8501	320.3857	0.5892
	离散 Coons	0.0247	71	11.0331	**0.0028**	0.8501	320.0014	**0.5888**
	前向 Laplacian	0.1690	**53**	**7.9965**	**0.0028**	**0.8503**	**317.5714**	0.5898

接下来, 我们在一个更复杂的示例中考察不同初始参数化对参数化结果的影响. 为了检查需要迭代多少步才能收敛到相同的精度水平, 我们在这个示例中将最大迭代



(a) 相同点方法 (b) 离散 Coons 方法 (c) 前向 Laplacian 方法

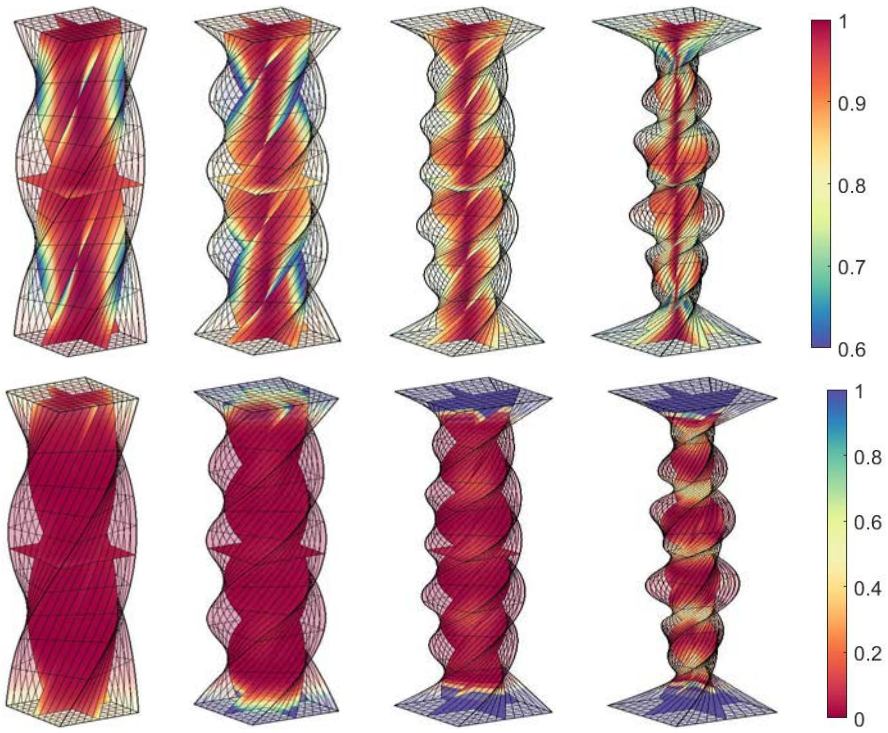
图 4.6 鸭子: 由不同初始参数化得到的参数化结果
Fig. 4.6 Duck: Parameterization Results from Different Initializations

次数 max_iter 设置为 500. 如图 4.6 所示, 同样地, 使用不同初始化方法得到的参数化质量几乎相同. 这表明从不同的初始参数化出发, 我们的方法收敛到了相同的最小值. 这也说明了本章所提出方法的鲁棒性.

尽管不同的初始参数化对结果参数化的质量影响有限, 但在迭代次数和计算成本上存在较大差异. 如表 4.1 所示, 相同点方法在初始化计算成本方面表现最好. 然而, 在后续优化过程中需要更多的迭代次数, 因此总体计算时间上表现更差. 相反, 前向 Laplacian 方法在初始化过程中的计算成本方面表现最差, 但在整体计算时间上表现最好. 因此, 我们建议采用前向 Laplacian 方法作为我们体参数化方法的初始化方法.

4.4.3 鲁棒性测试

在本小节中, 我们进一步测试所提出方法的鲁棒性. 考虑由每个方向有 6 个控制点的三次 NURBS 体参数化的长方体. 如图 4.7 所示, 我们将第 k 个平面的控制点按照 $k \cdot \theta^{\circ}/5$ $(k = 0, 1, \cdots, 5)$ 进行旋转. 我们称这些模型为 Rotθ. 随着扭转角度 θ 的增加, 计算域的几何形状也变得越来越复杂.



(a) Rot180 (b) Rot360 (c) Rot540 (d) Rot720

图 4.7 分别扭转一个长方体 180°, 360°, 540°, and 720° 得到的参数化结果

Fig. 4.7 Parameterization Results Obtained by Rotating a Rectangular Prism 180°, 360°, 540°, and 720°

表 4.2 将长方体扭转不同角度得到的体参数化结果的质量统计.

Tab. 4.2 Statistics of Parameterizations Obtained by Rotating a Rectangular Prism at Different Angles

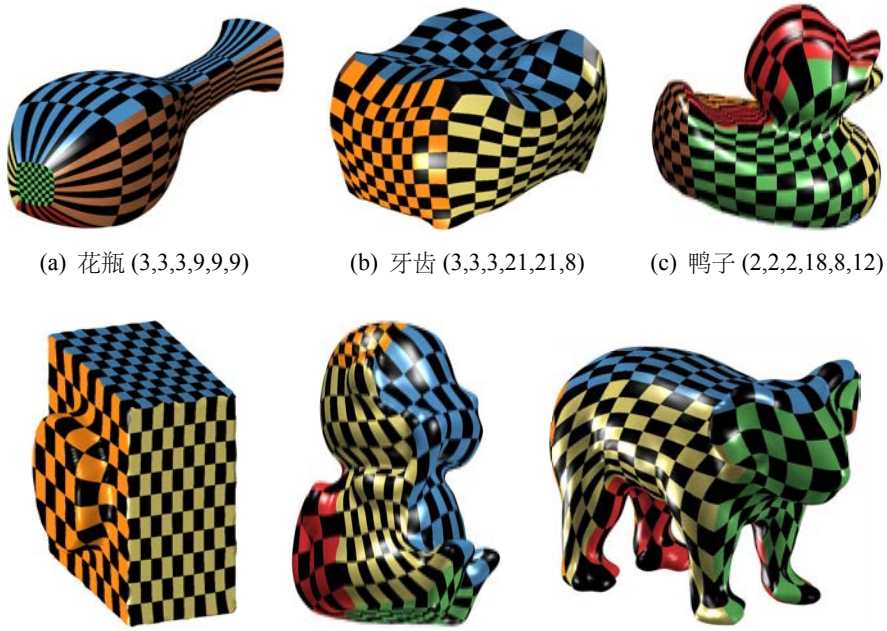
模型	迭代次数	计算时间 (秒)	比例雅可比		均匀性	
			最小值	平均值	最大值	平均值
Rot180	16	0.2734	0.5522	0.8986	0.6047	0.0310
Rot360	22	0.3376	0.4998	0.8560	2.0608	0.0732
Rot540	49	0.6907	0.6230	0.8811	9.4169	0.3602
Rot720	84	1.1965	0.5072	0.8458	67.1141	1.8257

从表 4.2 可以看出, 随着扭转角度 θ 的增加, 迭代次数和计算时间也都增加了, 所得到的参数化质量也相应变差. 不过可以看到所有模型的比例雅可比的最小值都大于 0.0, 这说明所有得到的体参数化结果都是单值的. 此外, 所有的参数化结果都通过了严格的单值性检验测试. 这进一步验证了本章方法的鲁棒性.

4.4.4 与高精度 Gauss 积分方案对比

在第 4.3.2 节中, 我们介绍了一种减缩积分策略. 这无疑会提高本章所提出体参数化方法的计算效率, 然而, *该减缩积分策略是否会导致精度上的损失?*

为了比较所提出的减缩积分策略和高精度 Gauss 积分之间的差异, 我们考虑一些具有更复杂几何的模型. 模型的边界表示和几何信息如图 4.8 所示, 其中括号内的数字代表模型的次数 p, q, r 和每个方向上控制顶点的个数 n, m, l. 这里高精度 Gauss 积分是指每个单元上采用 $(p+1)(q+1)(r+1)$ 个 Gauss 积分点.



(a) 花瓶 (3,3,3,9,9,9) (b) 牙齿 (3,3,3,21,21,8) (c) 鸭子 (2,2,2,18,8,12)

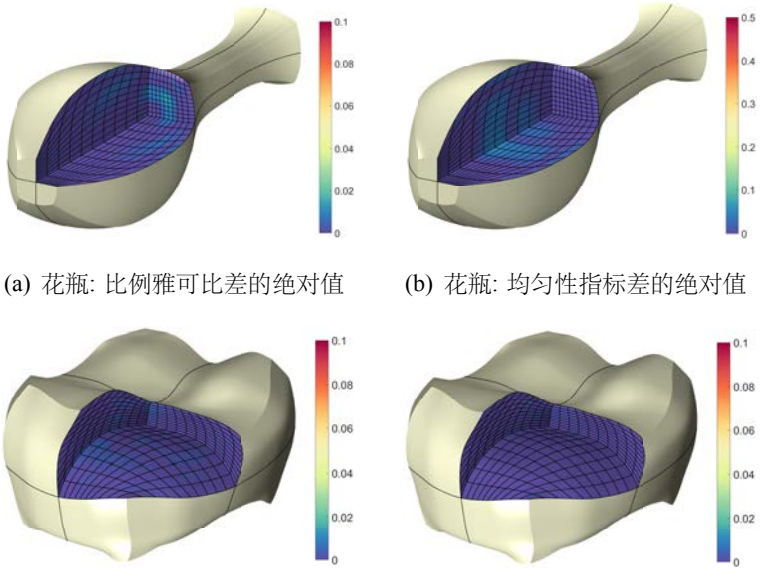
(d) 元器件 (3,25) (e) 猴子 (3,25) (f) 考拉 (3,25)

图 4.8 六个复杂模型的边界表示及其几何信息 (p,q,r,n,m,l)

Fig. 4.8 Boundary Representations and Geometric Information (p,q,r,n,m,l) of Six Complex Models

这两种积分方案生成的 NURBS 体参数结果的比例雅可比和均匀性指标差的绝对值如图 4.9 所示. 可以看到两种方法所得到的参数化结果的质量指标差的绝对值非常接近 0.0, 这意味着在参数化质量方面, 缩减积分策略和高精度 Gauss 积分之间没有显着差异. 更详细的统计数据可以在表 4.4 中每个模型的第一行和最后一行找到.

就计算时间而言, 如图 4.10 所示, 本章所提出的缩减积分策略极大地降低了计算成本. 虽然在牙齿, 鸭子和考拉模型中, 高精度数值积分方案所需要的迭代次数略少于减缩积分策略, 但该方法每次迭代的计算成本要高得多. 值得注意的是, 在后面三个规模更大的模型中, 高精度数值积分所需的计算时间是减缩积分方案的三到四倍. 因此, 所提出的缩减积分策略在没有显着损失参数化质量的情况下有效地降低了计算成本.



(a) 花瓶: 比例雅可比差的绝对值　　(b) 花瓶: 均匀性指标差的绝对值

(c) 牙齿: 比例雅可比差的绝对值　　(d) 牙齿: 均匀性指标差的绝对值

图 4.9　减缩积分方案与高精度数值积分方案的参数化结果对比

Fig. 4.9　Comparison of Parameterization Results between Reduced Integration Scheme and High-Accuracy Numerical Integration Scheme

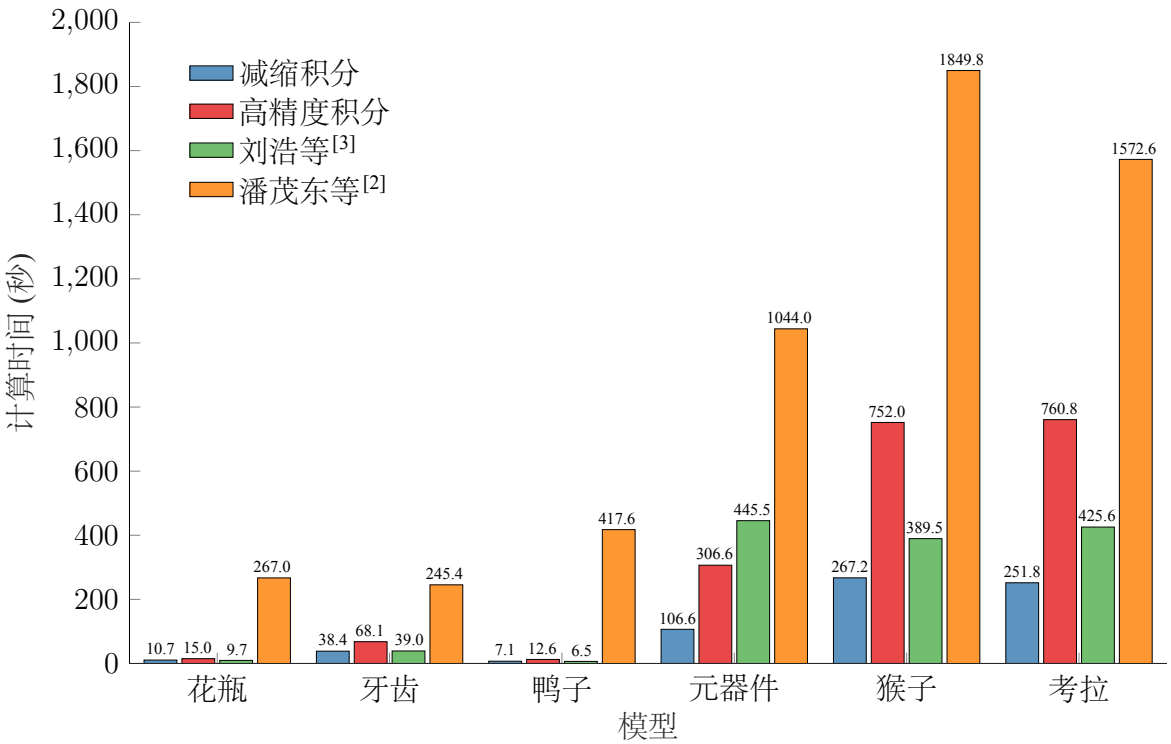
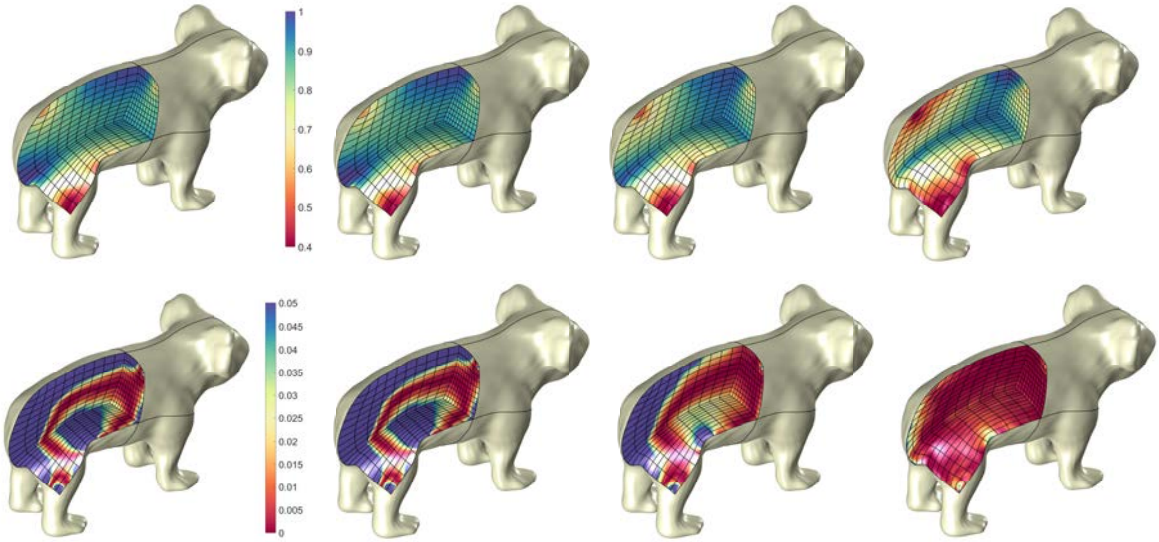


图 4.10　不同参数化方法的计算时间 (秒)

Fig. 4.10　Computation Time (in seconds) for Different Parameterization Methods

4.4.5 平衡参数 λ_{mips} 和 λ_{vol} 对参数化质量的影响

为了进一步探究平衡参数 λ_{mips} 和 λ_{vol} 对参数化结果的影响, 我们在考拉模型上进行测试. 具体来讲, 我们将参数 λ_{mips} 设置为 4, 并调整参数 λ_{vol} 的值, 以改变 λ_{mips} 和 λ_{vol} 的比例关系. 参数化结果如图 4.11 所示.



(a) $\lambda_{\text{mips}} = 4, \lambda_{\text{vol}} = 0$ (b) $\lambda_{\text{mips}} = 4, \lambda_{\text{vol}} = 1$ (c) $\lambda_{\text{mips}} = 4, \lambda_{\text{vol}} = 10$ (d) $\lambda_{\text{mips}} = 4, \lambda_{\text{vol}} = 100$

图 4.11 不同平衡参数 λ_{mips} 和 λ_{vol} 对参数化结果的影响

Fig. 4.11 Role of Different Balancing Parameters λ_{mips} and λ_{vol} on Parameterization Results

从图上可以看出, 所有不同参数组合均得到了适合分析的参数化结果, 这进一步说明了本章所提出的参数化方法的鲁棒性. 随着 λ_{vol} 的增加, 可以观察到体参数化的均匀性提高而角度扭曲增大. 更详细的统计数据列在了表 4.3 中.

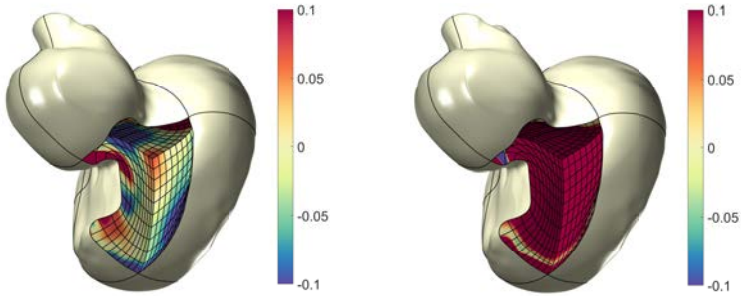
表 4.3 考拉模型上不同平衡参数 λ_{mips} 和 λ_{vol} 组合得到的参数化质量统计

Tab. 4.3 Statistics for Different Balancing Parameters λ_{mips} and λ_{vol} on the Koala Model

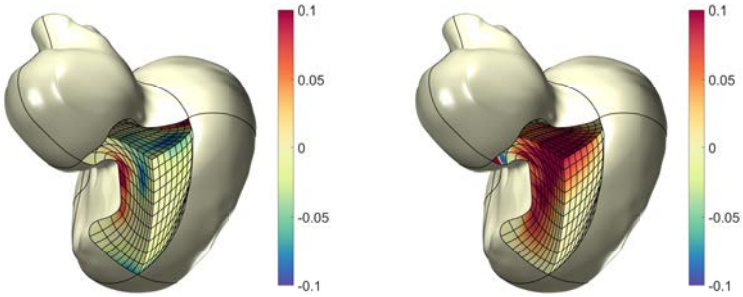
平衡参数 $\lambda_{\text{mips}}, \lambda_{\text{vol}}$	迭代次数	计算时间 (秒)	比例雅可比		均匀性	
			最小值	平均值	最大值	平均值
$\lambda_{\text{mips}} = 4, \lambda_{\text{vol}} = 0$	93	250.4371	**0.0003**	**0.7679**	865.5017	0.6159
$\lambda_{\text{mips}} = 4, \lambda_{\text{vol}} = 1$	94	251.2312	**0.0003**	0.7667	829.2130	0.5836
$\lambda_{\text{mips}} = 4, \lambda_{\text{vol}} = 10$	**84**	**208.3479**	**0.0003**	0.7550	612.6201	0.3923
$\lambda_{\text{mips}} = 4, \lambda_{\text{vol}} = 100$	100	268.3442	**0.0003**	0.6829	**79.4614**	**0.0511**

4.4.6 与现有方法的对比分析

在本节中, 我们进一步将本章方法与现有的方法进行比较. 正如文献[2]所指出的, 潘茂东等人的方法比文献[80]和[54]的方法更具优势. 因此, 本章将所提出的参数化方法与潘茂东等人方法[2] 和刘浩等人方法[3] 进行比较.



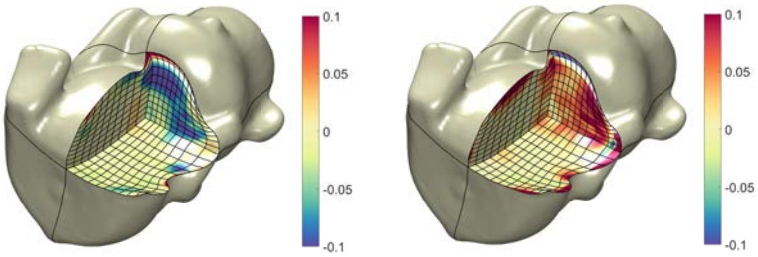
(a) 比例雅可比差的绝对值[2]　　(b) 均匀性指标差的绝对值[2]



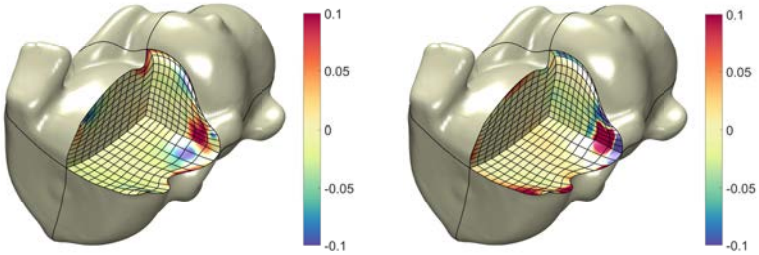
(c) 比例雅可比差的绝对值[3]　　(d) 均匀性指标差的绝对值[3]

图 4.12　鸭子: 与潘茂东等[2] 和刘浩等[3] 方法的对比

Fig. 4.12　Duck: Comparison with Pan et al.[2] and Liu et al.[3]



(a) 比例雅可比差的绝对值[2]　　(b) 均匀性指标差的绝对值[2]



(c) 比例雅可比差的绝对值[3]　　(d) 均匀性指标差的绝对值[3]

图 4.13　猴子: 与潘茂东等[2] 和刘浩等[3] 方法的对比

Fig. 4.13　Monkey: Comparison with Pan et al.[2] and Liu et al.[3]

首先需要说明的是, 潘茂东等人的方法[2] 对不同的模型需要不断地调整所涉及到的参数. 相比之下, 本章所提出的方法只需要调整少量参数, 而默认参数 ($\lambda_{\text{mips}} = 4$, $\lambda_{\text{vol}} = 10$) 在所有模型中都表现良好. 这可以看作是我们的方法相较于潘茂东等人方法的第一个优势.

表 4.4　不同方法所构造参数化结果的量化统计数据
Tab. 4.4　Quantitative Data for Parameterization Results Obtained by Different Methods

模型	方法	迭代次数	时间 (秒)	比例雅可比		均匀性	
				最小值	平均值	最大值	平均值
花瓶	高精度积分	**100**	15.0185	0.2856	0.8348	22.5510	0.6746
	潘茂东等[2]	-	267.0000	0.2673	0.7244	27.6335	0.9338
	刘浩等[3]	-	**9.6523**	0.2517	0.8226	**11.5626**	**0.4760**
	减缩积分	**100**	10.6535	**0.2883**	**0.8358**	22.5677	0.6738
牙齿	高精度积分	**63**	68.0538	**0.2603**	0.7447	**58.8538**	0.0844
	潘茂东等[2]	-	245.4000	**0.2603**	**0.7624**	58.8619	0.1490
	刘浩等[3]	-	39.0464	0.2500	0.7285	**58.8538**	0.0847
	减缩积分	64	**38.4208**	**0.2603**	0.7447	**58.8538**	**0.0843**
鸭子	高精度积分	**56**	12.6467	**0.0028**	0.8203	177.6282	0.3006
	潘茂东等[2]	-	417.6000	**0.0028**	**0.8532**	437.1407	1.3951
	刘浩等[3]	-	**6.5298**	**0.0028**	0.8425	221.0949	0.4515
	减缩积分	57	7.1310	**0.0028**	0.8202	**176.5330**	**0.2997**
元器件	高精度积分	**38**	306.5912	**0.0093**	0.9764	13.2030	0.0311
	潘茂东等[2]	-	1044.0000	**0.0093**	**0.9818**	45.4164	0.1112
	刘浩等[3]	-	445.5320	**0.0093**	0.9804	21.2408	0.0589
	减缩积分	**38**	**106.6206**	**0.0093**	0.9765	**12.8591**	**0.0305**
猴子	高精度积分	**100**	751.9515	**0.0034**	0.8440	134.4528	**0.1168**
	潘茂东等[2]	-	1849.8000	**0.0034**	0.8534	231.8213	0.2369
	刘浩等[3]	-	389.4500	**0.0034**	**0.8588**	165.5855	0.1536
	减缩积分	**100**	**267.2438**	**0.0034**	0.8447	**133.5604**	0.1188
考拉	高精度积分	**93**	760.8079	**0.0003**	0.7553	616.6197	0.3926
	潘茂东等[2]	-	1572.6000	-0.0169	0.7323	205.3992	0.3212
	刘浩等[3]	-	425.5910	**0.0003**	**0.7645**	**488.9502**	**0.3697**
	减缩积分	94	**251.8054**	**0.0003**	0.7550	612.6201	0.3923

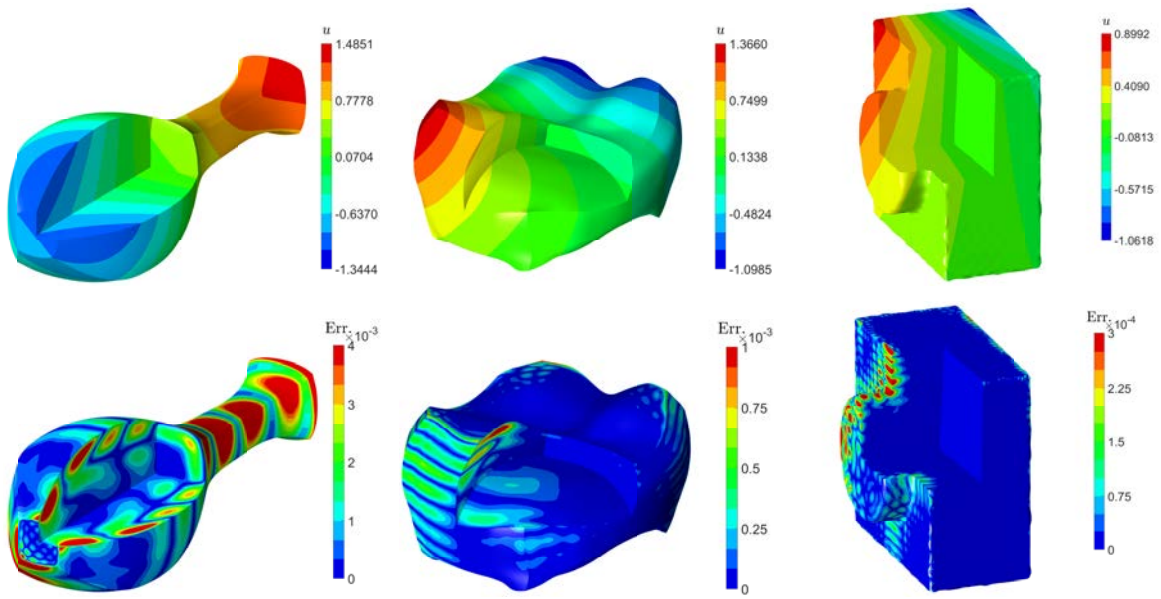
在三维模型中, 不同参数化结果之间的差异通常不容易观察到. 为此, 我们在本节

的图中展示不同方法之间比例雅可比和均匀性指标差的绝对值. 图 4.12 和图 4.13 分别展示了本章方法所构造的体参数化结果, 模型上的颜色代表本章方法与现有方法质量指标差的绝对值. 其中红色的区域代表本章方法具有更低的角度或体积扭曲.

不同方法所构造的参数化结果的质量指标统计见表 4.4. 本章算法与刘浩等人的方法[3] 表现相似, 在所有的模型上均成功地构造出单值的体参数化结果. 而潘茂东等人的方法[2] 未能在考拉模型上获得单值的参数化结果.

本章方法的另一个主要优势是计算效率. 尽管刘浩等人的方法[3] 采用纯 C++ 实现, 本章方法在前三个较小规模的模型上与该方法表现相近, 而在后面三个更大规模的模型中展现出了更高的计算效率. 这要归功于本章所提出的减缩积分策略. 此外, 我们的方法比刘浩等人的方法[3] 更加简洁, 易于实现.

与徐岗等人的方法[80] 相比, 潘茂东等人的方法[2] 采用了一个配点策略. 该策略大大减少了约束的数量并部分提高了计算效率. 另外, 与本论文第 3 章的思想类似, 潘茂东等人的方法[2] 尝试首先构造一个已经是双射的参数化, 然后通过极小化带有障碍函数的目标函数来进一步改善参数化质量, 以避免无效的参数化结果. 这些额外的消除折叠步通常需要过度优化, 而且对改善参数化质量贡献不大, 因此效率不高. 相比之下, 在本章方法中, 消除折叠和参数化质量提升同时进行. 这一特性极有可能带来效率上的提升. 最后, 虽然在潘茂东等人的实现中使用了商用优化求解器 Artelys Knitro[130] 和纯 C++ 实现, 但在所有模型中, 我们的方法仍然效率更高. 而在鸭子模型中, 本章方法比该方法快 50 倍左右.



(a) 花瓶, L_2误差 $= 1.72 \times 10^{-3}$　(b) 牙齿, L_2误差 $= 1.94 \times 10^{-4}$　(c) 元器件, L_2误差 $= 2.82 \times 10^{-5}$

图 4.14　本章方法构造的体参数化上求解 Poisson 方程

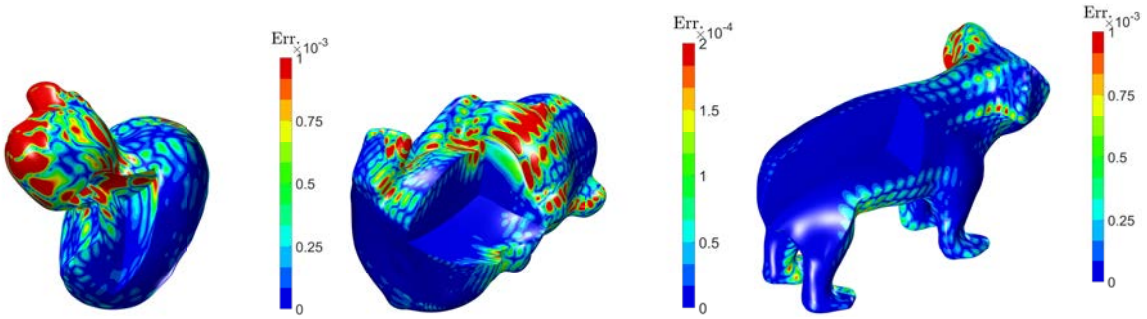
Fig. 4.14　Solving Poisson Equation on the Parametrization Constructed by Our Method

4.5 在等几何分析中的应用

本小节对所得到的参数化在求解三维 Poisson 方程中的适用性进行测试:

$$\begin{cases} -\Delta \mathscr{U}(\boldsymbol{x}) = f, & \boldsymbol{x} \in \Omega, \\ \mathscr{U}(\boldsymbol{x}) = g, & \boldsymbol{x} \in \partial\Omega, \end{cases} \tag{4.25}$$

其中, $\partial\Omega$ 表示计算域 Ω 的边界. 这里, 源项 $f = -e^{x+z}sin(y)$, 相应的精确解为 $\mathscr{U}^{\text{exact}} = e^{x+z}sin(y)$. 这里, Dirichlet 边界条件采用 L_2 投影方法施加.



(a) 本章方法, L_2 误差分别为 1.146×10^{-3}, 6.784×10^{-5}, 1.171×10^{-4} (从左到右)

(b) 潘茂东等方法[2], L_2 误差分别为 3.717×10^{-3}, 4.880×10^{-5}, 9.686×10^{-5} (从左到右)

(c) 刘浩等方法[3], L_2 误差分别为 1.093×10^{-3}, 8.646×10^{-5}, 1.205×10^{-4} (从左到右)

图 4.15 不同方法构造的体参数化上求解 Poisson 方程的绝对误差分布
Fig. 4.15 Absolute Error Distribution in Solving the Poisson Equation on Volumetric Parametrizations Constructed by Different Methods

图 4.14 展示了在花瓶、牙齿和元器件模型上的精确解, 绝对误差分布及 L_2 误差. 这表明本章方法生成的参数化结果适用于等几何分析应用.

图 4.15 展示了在不同方法构造的参数化结果上求解 Poisson 方程 (4.25) 的绝对误差分布. 有趣的是, 在考拉模型中, 尽管潘茂东等人的方法[2] 生成的参数化具有自相交, 但仍然获得了良好的数值解. 通常情况下, 雅可比行列式为负或者接近于 0 的区域出现在边界附近. 因此, 在这个例子中, Gauss 积分点恰好没有落在这些区域上. 然而, 这种具有自相交的参数化在细化过程中仍然存在数值不稳定的隐患.

最后, 我们强调, 等几何分析求解误差不仅与参数化的扭曲程度有关, 还与方程本身的特性有关. 换言之, 扭曲程度较小或具有良好正交均匀性的参数化未必会对所有的偏微分方程产生更准确的数值解. 这也是为什么一些工作关注自适应等几何分析参数化, 特别是 r 自适应方法[52,53,131]. 在第 7 章中, 我们将进一步探讨这一问题. 尽管如此, 双射性仍然是实现适合分析参数化和良好数值稳定性的基本前提条件.

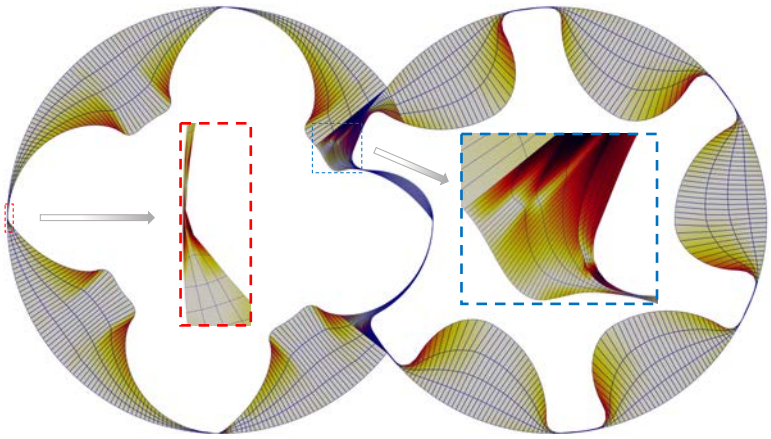
4.6 本章小结

本章提出了一种基于惩罚函数的体参数化方法. 通过引入均匀能量泛函, 进一步降低了参数化结果的体积扭曲. 同时, 本章提出了一种简单有效的惩罚函数, 并利用雅可比矩阵正则化技术给出了一个修正的加权目标泛函. 此外, 推导出了目标泛函的全解析梯度, 以提高一阶数值优化算法的数值稳定性. 为提高本章方法的计算效率, 开发了一种减缩数值积分策略. 数值实验表明, 本章所提出的体参数化方法与现有方法相比具有更加良好的鲁棒性和计算效率. 等几何分析仿真结果表明了本章方法所构造的体参数化适用于等几何分析求解.

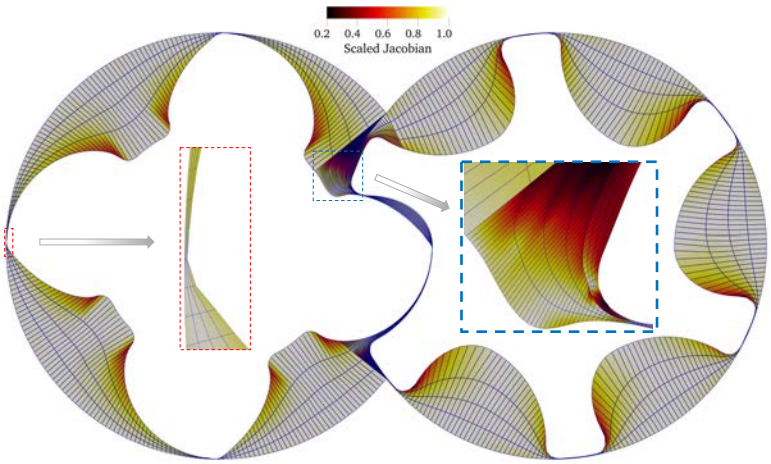
5 改进的椭圆参数化方法与预条件 Anderson 加速算法

5.1 引言

调和映射理论由于其优良的数学性质和扎实的理论基础在平面参数化问题上受到了广泛的应用和关注. 为了逼近调和映射的逆映射, 学术界已经发展出了多种计算方法, 例如变分调和方法[57], 最小二乘拟合[58,59]、最小化 (修正的)Winslow 函数[52,127]、以及基于 PDE 的椭圆网格生成 (EGG) 方法[72,74]. 实际上, 我们在第 3 章和第 4 章中所介绍的方法本质上也是在逼近调和映射的逆映射.



(a) 惩罚函数方法



(b) 椭圆网格生成 (EGG) 方法

图 5.1 螺杆机械截面: 罚函数方法[132] 和椭圆网格生成方法[72] 的参数化结果
Fig. 5.1 Twin-Screw Machine Cross-Sections: Parameterizations Constructed by Penalty Function Method[132] and Elliptic Grid Generation Method[72]

尽管上述多种方法都致力于在有限维样条空间里近似逆调和映射, 但不同方法的

数值表现却可能截然不同. 以图 5.1 中展示的双螺杆压缩机切片为例, 该几何形状极具挑战性, 特别是两个转子间隙具有极端的纵横比. 许多参数化方法虽然在理论或基准几何上表现出色, 但在这种复杂的实际应用场景中却表现不尽人意. 图 5.1(a) 展示了由第 4 章中所介绍的惩罚函数方法[132] 所构造的参数化结果. 对于这种高度复杂的几何形状, 正确地选择相关参数变得尤为重要. 在我们的研究过程中, 还观察到参数域的尺寸也会对参数化结果产生显著影响, 这显然是不合理的. 此外, 罚函数方法在这个实例中收敛极其缓慢. 这是因为在所有迭代步上我们都不允许雅可比行列式的值小于零, 因此对这一几何形状, 每个迭代步的步长都非常的小. 与之对比, 基于偏微分方程 (PDE) 的 EGG 方法[72] 生成了更为满意的参数化结果, 如图 5.1(b) 所示.

在基于 PDE 的方法中, 需要求解一组非线性方程. 文献[72]建议使用 Newton 型求解器, 然而该方法在非线性迭代过程中需要频繁显式地 (如标准的 Newton 方法) 或隐式地 (如 Newton-Krylov 方法) 计算其雅可比矩阵, 对大规模问题而言计算代价高. 对于大规模问题来说, 计算代价极高. 这也意味着, 尽管这些方法在生成高质量的参数化上有优势, 但在计算效率方面还需进一步优化.

1962 年, Anderson[133,134] 开发了一种名为外推法的计算技术, 用于加速非线性不动点迭代的收敛. 这种技术如今在应用数学界被称为 Anderson 加速算法 (Anderson acceleration, 简称 AA). 与 Newton 型方法不同, AA 无需计算雅可比矩阵. 尽管 AA 的加速收敛已被许多研究人员广泛观察到, 但对该方法的收敛性分析结果直到最近才被报道. 2015 年, Toth 和 Kelly[135] 首次证明, 如果固定点映射是一个收缩映射, 那么无阻尼的平稳 AA(sAA) 在局部是 r 线性收敛的. 后来, 文献[136]将这一结果扩展到带阻尼因子的 AA 算法. 最近, 文献[137]将 sAA 应用于求解稳态不可压 Navier-Stokes 方程的 Picard 迭代中, 并证明 AA 可以提高 Picard 迭代的收敛速率. 张举勇等人[138,139] 最近将 AA 应用于计算机图形学中的几何处理问题.

在本章中, 我们提出了几种针对基于 PDE 的 EGG 参数化方法的高效求解策略. 基本思想是将非线性问题转化为固定点迭代问题, 然后应用 (预条件)AA 来加速收敛.

本章的主要贡献如下:

- EGG 的一个缺陷是单元尺寸往往会不均匀, 特别是在曲率变化较大的边界附近. 为了削弱其影响并提高参数化的质量, 我们在参数化控制方程中引入了一个缩放函数 $\mathbb{A}$, 并给出其在 Sobolev 空间 H^1 中的离散格式;

- 在非线性迭代过程中, 雅可比矩阵的计算至关重要. 为了提高计算效率和数值稳定性, 我们为离散格式推导出了解析的雅可比矩阵;

- 采用 AA 算法避免频繁更新雅可比矩阵以降低计算开销. 为提高其鲁棒性并加速收敛, 提出了一个预条件 AA 框架. 提出了三种预条件策略, 即对角雅可比预

条件子、块对角雅可比预条件子和全雅可比预条件子. 此外, 为降低每次迭代的计算成本, 我们每隔 N_{update} 步更新预条件子.

5.2 椭圆参数化构造方法

椭圆网格生成技术 (EGG) 是传统网格生成中的一种常用方法, 常常用于从给定的计算域边界描述出发生成结构化网格. Hinz 等人[72] 将这一方法引入到等几何分析适合分析的参数化问题中来. 其基本思想是通过求解如下的 Laplacian 方程, 来逼近参数域 $\hat{\Omega}$ 和计算域 Ω 之间的调和映射 $\boldsymbol{x}$:

$$\begin{cases} \Delta\xi(x,y)=0 \\ \Delta\eta(x,y)=0 \end{cases} \quad \text{s.t. } \boldsymbol{x}^{-1}|_{\partial\Omega}=\partial\hat{\Omega}. \tag{5.1}$$

问题 (5.1) 是一类特殊的 Dirichlet 问题, 如果其边界 $\partial\Omega$ 满足 $C^{1,\alpha}$ Hölder 连续性条件, 则该问题的解存在, 其中 $\alpha\in(0,1)$, 且解的唯一性由极大值原理保证. 由于参数域 $\hat{\Omega}$ 被假定为凸的 (通常为单位正方形), 根据 Radó-Kneser-Choquet 定理[120], 该问题的唯一解 $\boldsymbol{x}^{-1}$ 提供了一个参数域 $\hat{\Omega}$ 与计算域 Ω 间的一一映射, 即雅可比行列式 $|\mathcal{J}|$ 在参数域内部不为零.

根据第一变分原理, 求解带有特定边界条件的方程 $\Delta\boldsymbol{\xi}=0$ 等价于找到一个映射 $\boldsymbol{x}^{-1}$, 使得该映射 $\boldsymbol{x}^{-1}$ 具有最小的 Dirichlet 能量

$$\mathscr{E}^{\text{Dirichlet}}=\frac{1}{2}\int_{\Omega}\left(\left(\frac{\partial\xi}{\partial x}\right)^2+\left(\frac{\partial\xi}{\partial y}\right)^2+\left(\frac{\partial\eta}{\partial x}\right)^2+\left(\frac{\partial\eta}{\partial y}\right)^2\right)d\Omega \tag{5.2}$$

且满足相同的边界条件.

将式 (5.2) 中的积分从计算域 Ω 变换到参数域 $\hat{\Omega}$, 我们得到:

$$\mathscr{E}^{\text{Winslow}}=\frac{1}{2}\int_{\hat{\Omega}}\frac{x_\xi^2+x_\eta^2+y_\xi^2+y_\eta^2}{|\mathcal{J}|}d\hat{\Omega}. \tag{5.3}$$

这就是著名的 Winslow 泛函. 在第 3 章中, 我们曾利用该能量泛函构造平面区域上的 NURBS 参数化. 这意味着在 Sobolev 空间 H^1 中, 求解问题 (5.1) 和找到 Winslow 能量泛函 (5.3) 的最小值本质上是等价的.

5.2.1 Sobolev 空间 H^2 中的离散格式

在构造适合分析的参数化时, 往往对从参数域 $\hat{\Omega}$ 到计算域 Ω 的映射 $\boldsymbol{x}$ 更感兴趣. 为此, 将 Laplacian 方程组 (5.1) 转换成如下的非线性向量值二阶 PDE 问题:

$$\begin{cases} \mathscr{L}x = 0 \\ \mathscr{L}y = 0 \end{cases} \quad \text{s.t. } \boldsymbol{x}|_{\partial\hat{\Omega}} = \partial\Omega, \tag{5.4}$$

其中

$$\mathscr{L} = g_{22}\frac{\partial^2}{\partial\xi^2} - 2g_{12}\frac{\partial^2}{\partial\xi\partial\eta} + g_{11}\frac{\partial^2}{\partial\eta^2} \tag{5.5}$$

是一个微分算子, $g_{11} = \boldsymbol{x}_{\boldsymbol{\xi}} \cdot \boldsymbol{x}_{\xi}$, $g_{12} = \boldsymbol{x}_{\xi} \cdot \boldsymbol{x}_{\boldsymbol{\eta}}$, $g_{22} = \boldsymbol{x}_{\boldsymbol{\eta}} \cdot \boldsymbol{x}_{\boldsymbol{\eta}}$ 是度量张量 $\boldsymbol{\mathcal{G}}$ 的元素.

为保证更好的收敛性, Hinz 等人[73] 将式 (5.5) 中的微分算子 $\mathscr{L}$ 替换为:

$$\widetilde{\mathscr{L}} = \frac{\mathscr{L}}{g_{11} + g_{22}}. \tag{5.6}$$

上述算子 (5.6) 对具有不同比例几何的计算域具有更加一致的收敛准则, 且在数值实验中具有更好的收敛性质. 因此, 在本章中, 我们采用上述算子 (5.6).

记 $\mathbb{S}$ 为体参数化中 NURBS 基函数所张成的样条空间. 令 $\mathbb{S}_0 = \{R_i \in \mathbb{S} : R_i|_{\partial\hat{\Omega}} = 0\}$ 表示在参数域边界 $\partial\hat{\Omega}$ 上为零的 $R_i \in \mathbb{S}$ 的集合. 将集合 $\mathbb{S}_0$ 中的元素作为测试函数, 有如下变分形式:

$$\forall R_i \in \mathbb{S}_0 : \begin{cases} \boldsymbol{\mathcal{F}}^x = \mathbf{0}, \\ \boldsymbol{\mathcal{F}}^y = \mathbf{0}, \end{cases} \quad \text{s.t. } \boldsymbol{x}|_{\partial\hat{\Omega}} = \partial\Omega, \tag{5.7}$$

其中

$$\boldsymbol{\mathcal{F}}^x = \int_{\hat{\Omega}} \mathbf{R}\, \widetilde{\mathscr{L}}x \, d\hat{\Omega},$$
$$\boldsymbol{\mathcal{F}}^y = \int_{\hat{\Omega}} \mathbf{R}\, \widetilde{\mathscr{L}}y \, d\hat{\Omega},$$

$\mathbf{R}$ 表示由 NURBS 基函数 $R_i \in \mathbb{S}_0$ 组成的列向量. 接下来, 以已知的边界控制顶点作为 Dirichlet 边界条件, 未知的内部控制顶点可以通过求解上述非线性方程组 (5.7) 得到.

5.2.2 H^2 离散方法的解析雅可比计算

解析雅可比矩阵不仅对计算效率有着举足轻重的作用, 同时对数值稳定性也具有重要意义. 在本节中, 我们将推导非线性方程组 (5.7) 雅可比矩阵的解析表达式.

为了符号上的清晰, 首先我们规定 $\partial_{\mathbf{P}^x} = \partial/\partial\mathbf{P}^x$ 和 $\partial_{\mathbf{P}^y} = \partial/\partial\mathbf{P}^y$ 分别表示对控制顶点 $\mathbf{P}$ 的所有 x 坐标分量和所有 y 坐标分量的偏导数. 此外, 我们用 $()_{,\alpha}$ 和 $()_{,\alpha\beta}$ $(\alpha = \xi, \eta, \beta = \xi, \eta)$ 分别表示对参数 ξ, η 的一阶和二阶偏导数.

对度量张量 $\boldsymbol{\mathcal{G}}$ 的各个元素 g_{ij} 求偏导数, 我们有:

$$\begin{aligned}
&\partial_{\mathbf{P}^x} g_{11} = 2x_\xi \mathbf{R}_{,\xi}, && \partial_{\mathbf{P}^y} g_{11} = 2y_\xi \mathbf{R}_{,\xi},\\
&\partial_{\mathbf{P}^x} g_{22} = 2x_\eta \mathbf{R}_{,\eta}, && \partial_{\mathbf{P}^y} g_{22} = 2y_\eta \mathbf{R}_{,\eta},\\
&\partial_{\mathbf{P}^x} g_{12} = x_\eta \mathbf{R}_{,\xi} + x_\xi \mathbf{R}_{,\eta}, && \partial_{\mathbf{P}^y} g_{12} = y_\eta \mathbf{R}_{,\xi} + y_\xi \mathbf{R}_{,\eta}.
\end{aligned} \tag{5.8}$$

利用导数的乘积规则并代入上述公式 (5.8), 我们有:

$$\begin{aligned}
&\partial_{\mathbf{P}^x} \mathscr{L} x = x_{,\xi\xi} \partial_{\mathbf{P}^x} g_{22} - 2x_{,\xi\eta} \partial_{\mathbf{P}^x} g_{12} + x_{,\eta\eta} \partial_{\mathbf{P}^x} g_{11} + g_{22} \mathbf{R}_{,\xi\xi} - 2g_{12} \mathbf{R}_{,\xi\eta} + g_{11} \mathbf{R}_{,\eta\eta},\\
&\partial_{\mathbf{P}^y} \mathscr{L} y = y_{,\xi\xi} \partial_{\mathbf{P}^y} g_{22} - 2y_{,\xi\eta} \partial_{\mathbf{P}^y} g_{12} + y_{,\eta\eta} \partial_{\mathbf{P}^y} g_{11} + g_{22} \mathbf{R}_{,\xi\xi} - 2g_{12} \mathbf{R}_{,\xi\eta} + g_{11} \mathbf{R}_{,\eta\eta},\\
&\partial_{\mathbf{P}^y} \mathscr{L} x = x_{,\xi\xi} \partial_{\mathbf{P}^y} g_{22} - 2x_{,\xi\eta} \partial_{\mathbf{P}^y} g_{12} + x_{,\eta\eta} \partial_{\mathbf{P}^y} g_{11},\\
&\partial_{\mathbf{P}^x} \mathscr{L} y = y_{,\xi\xi} \partial_{\mathbf{P}^x} g_{22} - 2y_{,\xi\eta} \partial_{\mathbf{P}^x} g_{12} + y_{,\eta\eta} \partial_{\mathbf{P}^x} g_{11}.
\end{aligned} \tag{5.9}$$

接着可以计算出缩放后的非线性方程组 (5.6) 的偏导数如下:

$$\begin{aligned}
\partial_{\mathbf{P}^x} \widetilde{\mathscr{L}} x &= \frac{\partial_{\mathbf{P}^x} \mathscr{L} x - \widetilde{\mathscr{L}} x \left(\partial_{\mathbf{P}^x} g_{11} + \partial_{\mathbf{P}^x} g_{22}\right)}{g_{11} + g_{22}},\\
\partial_{\mathbf{P}^y} \widetilde{\mathscr{L}} x &= \frac{\partial_{\mathbf{P}^y} \mathscr{L} x - \widetilde{\mathscr{L}} x \left(\partial_{\mathbf{P}^y} g_{11} + \partial_{\mathbf{P}^y} g_{22}\right)}{g_{11} + g_{22}},\\
\partial_{\mathbf{P}^x} \widetilde{\mathscr{L}} y &= \frac{\partial_{\mathbf{P}^x} \mathscr{L} y - \widetilde{\mathscr{L}} y \left(\partial_{\mathbf{P}^x} g_{11} + \partial_{\mathbf{P}^x} g_{22}\right)}{g_{11} + g_{22}},\\
\partial_{\mathbf{P}^y} \widetilde{\mathscr{L}} y &= \frac{\partial_{\mathbf{P}^y} \mathscr{L} y - \widetilde{\mathscr{L}} y \left(\partial_{\mathbf{P}^y} g_{11} + \partial_{\mathbf{P}^y} g_{22}\right)}{g_{11} + g_{22}}.
\end{aligned} \tag{5.10}$$

最后, 我们得到非线性方程组 (5.7) 的解析雅可比矩阵为

$$jac(\boldsymbol{\mathcal{F}}) = \begin{bmatrix} \partial_{\mathbf{P}^x} \boldsymbol{\mathcal{F}}^x & \partial_{\mathbf{P}^x} \boldsymbol{\mathcal{F}}^y \\ \partial_{\mathbf{P}^y} \boldsymbol{\mathcal{F}}^x & \partial_{\mathbf{P}^y} \boldsymbol{\mathcal{F}}^y \end{bmatrix}, \tag{5.11}$$

其中

$$\begin{aligned}
&\partial_{\mathbf{P}^x} \boldsymbol{\mathcal{F}}^x = \int_{\hat{\Omega}} \mathbf{R}\, \partial_{\mathbf{P}^x} \widetilde{\mathscr{L}} x \, d\hat{\Omega}, && \partial_{\mathbf{P}^y} \boldsymbol{\mathcal{F}}^x = \int_{\hat{\Omega}} \mathbf{R}\, \partial_{\mathbf{P}^y} \widetilde{\mathscr{L}} x \, d\hat{\Omega},\\
&\partial_{\mathbf{P}^x} \boldsymbol{\mathcal{F}}^y = \int_{\hat{\Omega}} \mathbf{R}\, \partial_{\mathbf{P}^x} \widetilde{\mathscr{L}} y \, d\hat{\Omega}, && \partial_{\mathbf{P}^y} \boldsymbol{\mathcal{F}}^y = \int_{\hat{\Omega}} \mathbf{R}\, \partial_{\mathbf{P}^y} \widetilde{\mathscr{L}} y \, d\hat{\Omega}.
\end{aligned} \tag{5.12}$$

5.2.3 Sobolev 空间 H^1 中的缩放离散格式

图 5.2 左侧展示了由非线性方程组 (5.7) 生成的参数化结果. 图中颜色代表比例雅可比值, 其中值为 1 表示参数化具有最佳的正交性. 可以观察到, 红色圆圈区域内

存在一些不均匀单元. 这种现象是求解式 (5.4) 中的 Laplacian 方程生成网格的固有特性[57]. 文献[72]指出, 这一缺陷可以通过对当前参数化进行细化并重新求解非线性方程组 (5.7) 来改善, 从而更好地逼近调和映射. 然而, 这种细化操作会引入不必要的控制顶点, 导致存储 CAD 几何模型所需要的内存成倍增加, 从而对后续分析和其他几何处理带来不必要的困扰. 因此, 我们希望在不增加控制顶点数量的基础之上, 改善所得到参数化的质量.

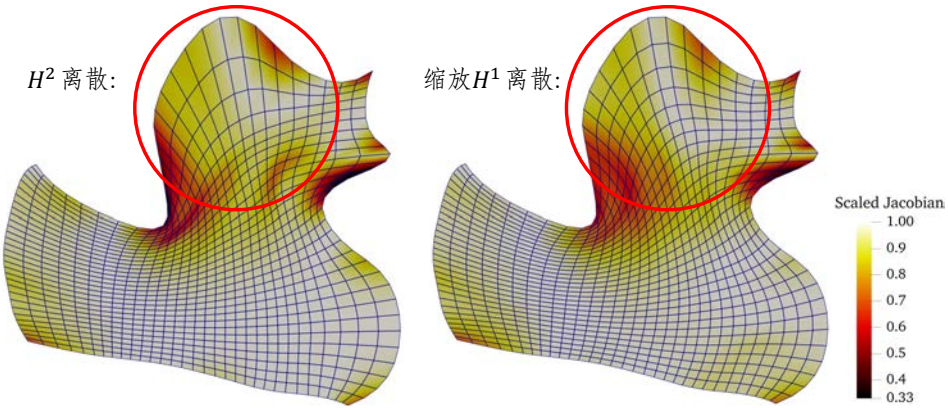


图 5.2　鸭子: H^2 离散参数化结果 (左) 和缩放 H^1 离散参数化结果 (右)

Fig. 5.2　Duck: Parameterization by H^2 Discretization (Left) and Scaled H^1 Discretization (Right).

为此, 我们在方程组 (5.1) 中引入一个缩放矩阵 $\mathbb{A}(\boldsymbol{x})$. 考虑如下方程

$$\begin{cases} \nabla\cdot(\mathbb{A}(\boldsymbol{x})\nabla\xi(\boldsymbol{x}))=0 \\ \nabla\cdot(\mathbb{A}(\boldsymbol{x})\nabla\eta(\boldsymbol{x}))=0 \end{cases} \quad \text{s.t. } \boldsymbol{x}^{-1}|_{\partial\Omega}=\partial\hat{\Omega}.$$

本章中, 取矩阵

$$\mathbb{A}=\begin{bmatrix} 1/|\mathcal{J}| & 0 \\ 0 & 1/|\mathcal{J}| \end{bmatrix}. \tag{5.13}$$

由变分原理, 其在 Sobolev 空间 H^1 中的等价变分问题为:

$$\forall R_i\in\mathbb{S}_0:\begin{cases} \boldsymbol{\mathcal{F}}^x_{H^1}=\mathbf{0}, \\ \boldsymbol{\mathcal{F}}^y_{H^1}=\mathbf{0}, \end{cases} \quad \text{s.t. } \boldsymbol{x}|_{\partial\hat{\Omega}}=\partial\Omega, \tag{5.14}$$

其中

$$\begin{aligned} \boldsymbol{\mathcal{F}}^x_{H^1}&=\int_{\hat{\Omega}}\nabla_{\boldsymbol{x}}\mathbf{R}\cdot\nabla_{\boldsymbol{x}}\xi\,d\hat{\Omega}, \\ \boldsymbol{\mathcal{F}}^y_{H^1}&=\int_{\hat{\Omega}}\nabla_{\boldsymbol{x}}\mathbf{R}\cdot\nabla_{\boldsymbol{x}}\eta\,d\hat{\Omega}, \end{aligned} \tag{5.15}$$

$\mathbf{R}$ 表示 NURBS 基函数 $R_i\in\mathbb{S}_0$ 按列排成的矩阵.

图 5.2 右侧展示了求解非线性方程组 (5.14) 所得到的参数化结果. 可以看到, 无论

是正交性还是均匀性都得到了显著改善. 最重要的一点是, 这一优化是在不增加控制顶点数量的条件下实现的.

注 5.1. 关于缩放矩阵 $\mathbb{A}(x)$ 的进一步讨论可以参考 Jochen Hinz 和 Annalisa Buffa 最近的文章 [140].

5.2.4 缩放 H^1 离散的解析雅可比计算

在下面的推导中, $\partial_{\mathbf{P}_j^x}$ 和 $\partial_{\mathbf{P}_j^y}$ 分别表示对第 j 个控制点 $\mathbf{P}_j$ 的 x 和 y 分量求偏导数. 首先, 我们计算雅可比矩阵 $\mathcal{J}$ 逆矩阵的偏导数:

$$\partial_{\mathbf{P}_j^x}\mathcal{J}^{-1} = -\mathcal{J}^{-1}\begin{bmatrix}\frac{\partial R_j}{\partial \xi}, 0\\ \frac{\partial R_j}{\partial \eta}, 0\end{bmatrix}\mathcal{J}^{-1}, \tag{5.16}$$

且

$$\partial_{\mathbf{P}_j^y}\mathcal{J}^{-1} = -\mathcal{J}^{-1}\begin{bmatrix}0, \frac{\partial R_j}{\partial \xi}\\ 0, \frac{\partial R_j}{\partial \eta}\end{bmatrix}\mathcal{J}^{-1}. \tag{5.17}$$

由于 $\nabla_{\boldsymbol{x}}\xi$ 和 $\nabla_{\boldsymbol{x}}\eta$ 仅仅是雅可比逆矩阵 $\mathcal{J}^{-1}$ 的第一列和第二列, 即

$$[\nabla_{\boldsymbol{x}}\xi, \nabla_{\boldsymbol{x}}\eta] = \begin{bmatrix}\frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x}\\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y}\end{bmatrix} = \mathcal{J}^{-1}, \tag{5.18}$$

我们有

$$\begin{aligned}\partial_{\mathbf{P}_j^x}\nabla_{\boldsymbol{x}}\xi &= \left(\partial_{\mathbf{P}_j^x}\mathcal{J}^{-1}\right)_{(:,1)}, \quad \partial_{\mathbf{P}_j^y}\nabla_{\boldsymbol{x}}\xi = \left(\partial_{\mathbf{P}_j^y}\mathcal{J}^{-1}\right)_{(:,1)},\\ \partial_{\mathbf{P}_j^x}\nabla_{\boldsymbol{x}}\eta &= \left(\partial_{\mathbf{P}_j^x}\mathcal{J}^{-1}\right)_{(:,2)}, \quad \partial_{\mathbf{P}_j^y}\nabla_{\boldsymbol{x}}\eta = \left(\partial_{\mathbf{P}_j^y}\mathcal{J}^{-1}\right)_{(:,2)},\end{aligned} \tag{5.19}$$

其中 $(\ )_{(:,i)}$ 表示矩阵 $(\ )$ 的第 i $(i=1,2)$ 列.

因此, 非线性方程组 (5.14) 的解析雅可比矩阵为

$$jac(\boldsymbol{\mathcal{F}}_{H^1}) = \begin{bmatrix}\partial_{\mathbf{P}^x}\boldsymbol{\mathcal{F}}_{H^1}^x & \partial_{\mathbf{P}^y}\boldsymbol{\mathcal{F}}_{H^1}^x\\ \partial_{\mathbf{P}^x}\boldsymbol{\mathcal{F}}_{H^1}^y & \partial_{\mathbf{P}^y}\boldsymbol{\mathcal{F}}_{H^1}^y\end{bmatrix}, \tag{5.20}$$

其中

$$\begin{aligned}(\partial_{\mathbf{P}^x}\boldsymbol{\mathcal{F}}_{H^1}^x)_{(i,j)} &= \int_{\hat{\Omega}} \partial_{\mathbf{P}_j^x}\mathcal{J}^{-1}\nabla_{\boldsymbol{\xi}}R_i\cdot\nabla_{\boldsymbol{x}}\xi + \nabla_{\boldsymbol{x}}R_i\cdot\partial_{\mathbf{P}_j^x}\nabla_{\boldsymbol{x}}\xi\, d\hat{\Omega},\\ (\partial_{\mathbf{P}^y}\boldsymbol{\mathcal{F}}_{H^1}^x)_{(i,j)} &= \int_{\hat{\Omega}} \partial_{\mathbf{P}_j^y}\mathcal{J}^{-1}\nabla_{\boldsymbol{\xi}}R_i\cdot\nabla_{\boldsymbol{x}}\xi + \nabla_{\boldsymbol{x}}R_i\cdot\partial_{\mathbf{P}_j^y}\nabla_{\boldsymbol{x}}\xi\, d\hat{\Omega},\\ (\partial_{\mathbf{P}^x}\boldsymbol{\mathcal{F}}_{H^1}^y)_{(i,j)} &= \int_{\hat{\Omega}} \partial_{\mathbf{P}_j^x}\mathcal{J}^{-1}\nabla_{\boldsymbol{\xi}}R_i\cdot\nabla_{\boldsymbol{x}}\eta + \nabla_{\boldsymbol{x}}R_i\cdot\partial_{\mathbf{P}_j^x}\nabla_{\boldsymbol{x}}\eta\, d\hat{\Omega},\\ (\partial_{\mathbf{P}^y}\boldsymbol{\mathcal{F}}_{H^1}^y)_{(i,j)} &= \int_{\hat{\Omega}} \partial_{\mathbf{P}_j^y}\mathcal{J}^{-1}\nabla_{\boldsymbol{\xi}}R_i\cdot\nabla_{\boldsymbol{x}}\eta + \nabla_{\boldsymbol{x}}R_i\cdot\partial_{\mathbf{P}_j^y}\nabla_{\boldsymbol{x}}\eta\, d\hat{\Omega}\end{aligned} \tag{5.21}$$

分别是非线性方程组 (5.14) 的雅可比矩阵的子块.

5.2.5 非线性方程组的等价固定点迭代问题

问题 (5.7) 和 (5.14) 最终都可以转化为如下的非线性方程组:

$$\boldsymbol{\mathcal{F}}(\mathbf{x}) = \mathbf{0}, \tag{5.22}$$

其中 $\mathbf{x} \in \mathbb{R}^n$, 且 $\boldsymbol{\mathcal{F}} : \mathbb{R}^n \to \mathbb{R}^n$.

对于上述问题, 典型的求解方法是 Newton 型方法, 例如文献[72]中所采用的截断 Newton 方法和伪时间步 Newton 方法. 然而, 在这些求解方法中, 每次迭代都必须计算和更新雅可比矩阵. 这对于大规模的问题来说无疑会大大增加计算负担. 为了缓解这一计算压力, 并降低每次迭代中的计算成本, 我们采取了一种更为高效的策略. 具体来说, 我们首先将非线性方程组 (5.22) 转化为如下的固定点迭代问题:

$$\mathbf{x} = \mathbf{x} + \boldsymbol{\mathcal{F}}(\mathbf{x}). \tag{5.23}$$

接下来, 我们采用 Anderson 加速 (AA) 算法来求解上述固定点迭代问题, 并加速收敛过程. 在传统的 Picard 迭代方法中, 每次更新仅依赖于上一次迭代的结果, 而 AA 算法则考虑了前 m 次迭代的信息, 通过对这些信息进行线性组合以最小化残差. 这里, m 被称为窗口大小, 我们记这种方法为 AA(m). 然而, 在研究中, 我们发现基本迭代方法 (5.23) 的收敛速度可能非常缓慢, 甚至对某些问题发散.

为进一步提高 AA 算法的数值稳定性和收敛速度, 本章提出一系列预条件 Anderson 加速策略. 具体地, 设 $\boldsymbol{\mathcal{M}}_k$ 为第 k 步的预条件子, 定义如下预条件固定点迭代:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \boldsymbol{\mathcal{M}}_k^{-1}\boldsymbol{\mathcal{F}}(\mathbf{x}_k). \tag{5.24}$$

5.3 预条件 Anderson 加速算法

在本节中, 我们将讨论如何选择合适的非线性预条件子 $\boldsymbol{\mathcal{M}}_k$.

5.3.1 非线性预条件子

设 $\mathbf{x}^*$ 是非线性方程组 $\boldsymbol{\mathcal{F}}(\mathbf{x}) = \mathbf{0}$ 的解, 而 $\mathbf{x}_k$ 是第 k 次迭代中 $\mathbf{x}^*$ 的估计, 且有 $\|\mathbf{e}_k\| = \|\mathbf{x}^* - \mathbf{x}_k\| \ll 1$.

将非线性方程组 $\boldsymbol{\mathcal{F}}(\mathbf{x})$ 在 $\mathbf{x}^*$ 附近 Taylor 展开, 我们有

$$\mathbf{0} = \boldsymbol{\mathcal{F}}(\mathbf{x}^*) = \boldsymbol{\mathcal{F}}(\mathbf{x}_k) + jac(\boldsymbol{\mathcal{F}}_k)\mathbf{e}_k + \frac{1}{2}\mathbf{E}_k^{\mathrm{T}}\mathbf{H}_k\mathbf{E}_k + \mathcal{O}(\|\mathbf{e}_k\|^3), \tag{5.25}$$

其中 $jac(\boldsymbol{\mathcal{F}}_k)$ 是第 k 步迭代中非线性方程组 $\boldsymbol{\mathcal{F}}(\mathbf{x})$ 的雅可比矩阵, $\mathbf{E}_k = diag\{\mathbf{e}_k^{\mathrm{T}}, \ldots, \mathbf{e}_k^{\mathrm{T}}\}$ 是由残差向量的转置 $\mathbf{e}_k^{\mathrm{T}}$ 组成的对角矩阵, 且

$$\mathbf{H}_k = diag\{\mathbf{H}^{(1)}(\mathbf{x}), \mathbf{H}^{(2)}(\mathbf{x}), \ldots, \mathbf{H}^{(n)}(\mathbf{x})\}$$

是由 $\mathbf{H}^{(i)}(\mathbf{x})$ 组成的块对角矩阵, 其中 $\mathbf{H}^{(i)}(\mathbf{x})$ 是在第 k 步迭代中第 i 个非线性方程 $\boldsymbol{\mathcal{F}}^{(i)}(x)$ $(i = 1, 2, \ldots, n)$ 的 Hessian 矩阵.

回忆预条件固定点迭代 (5.24), 我们有

$$\boldsymbol{\mathcal{F}}(\mathbf{x}_k) = \boldsymbol{\mathcal{M}}_k\left(\mathbf{x}_k - \mathbf{x}_{k+1}\right) = \boldsymbol{\mathcal{M}}_k\left((\mathbf{x}^* - \mathbf{x}_{k+1}) - (\mathbf{x}^* - \mathbf{x}_k)\right) = \boldsymbol{\mathcal{M}}_k\left(\mathbf{e}_{k+1} - \mathbf{e}_k\right). \quad (5.26)$$

将上式 (5.26) 代入 (5.25) 中, 我们得到

$$\begin{aligned} \mathbf{0} &= \boldsymbol{\mathcal{F}}(\mathbf{x}_k) + jac(\boldsymbol{\mathcal{F}}_k)\mathbf{e}_k + \frac{1}{2}\mathbf{E}_k^T\mathbf{H}_k\mathbf{E}_k + \mathcal{O}(\|\mathbf{e}_k\|^3) \\ &= \boldsymbol{\mathcal{M}}_k\left(\mathbf{e}_{k+1} - \mathbf{e}_k\right) + jac(\boldsymbol{\mathcal{F}}_k)\mathbf{e}_k + \frac{1}{2}\mathbf{E}_k^T\mathbf{H}_k\mathbf{E}_k + \mathcal{O}(\|\mathbf{e}_k\|^3). \end{aligned} \quad (5.27)$$

在上式 (5.27) 两边乘以预条件子的逆 $\boldsymbol{\mathcal{M}}_k^{-1}$, 我们得到

$$\begin{aligned} \mathbf{0} &= (\mathbf{e}_{k+1} - \mathbf{e}_k) + \boldsymbol{\mathcal{M}}_k^{-1}jac(\boldsymbol{\mathcal{F}}_k)\mathbf{e}_k + \frac{1}{2}\boldsymbol{\mathcal{M}}_k^{-1}\mathbf{E}_k^T\mathbf{H}_k\mathbf{E}_k + \mathcal{O}(\|\mathbf{e}_k\|^3) \\ &= \mathbf{e}_{k+1} - \left(\mathbf{I} - \boldsymbol{\mathcal{M}}_k^{-1}jac(\boldsymbol{\mathcal{F}}_k)\right)\mathbf{e}_k + \frac{1}{2}\boldsymbol{\mathcal{M}}_k^{-1}\mathbf{E}_k^T\mathbf{H}_k\mathbf{E}_k + \mathcal{O}(\|\mathbf{e}_k\|^3), \end{aligned} \quad (5.28)$$

其中 $\mathbf{I}$ 是 n 阶单位阵.

最终, 我们得到如下的误差估计

$$\mathbf{e}_{k+1} = \left(\mathbf{I} - \boldsymbol{\mathcal{M}}_k^{-1}jac(\boldsymbol{\mathcal{F}}_k)\right)\mathbf{e}_k - \frac{1}{2}\boldsymbol{\mathcal{M}}_k^{-1}\mathbf{E}_k^T\mathbf{H}_k\mathbf{E}_k + \mathcal{O}(\|\mathbf{e}_k\|^3). \quad (5.29)$$

如果令预条件子

$$\boldsymbol{\mathcal{M}}_k = jac(\boldsymbol{\mathcal{F}}_k), \quad (5.30)$$

那么式 (5.29) 中的误差估计变成

$$\mathbf{e}_{k+1} = -\frac{1}{2}jac(\boldsymbol{\mathcal{F}}_k)^{-1}\mathbf{E}_k^T\mathbf{H}_k\mathbf{E}_k + \mathcal{O}(\|\mathbf{e}_k\|^3). \quad (5.31)$$

也就是说, $\|\mathbf{e}_{k+1}\| = \mathcal{O}(\|\mathbf{e}_k\|^2)$.

这意味着, 如果我们从一个足够好的初始值出发, 并且在每次迭代中更新预条件子矩阵 $\boldsymbol{\mathcal{M}}_k$, 那么我们的预条件 AA 算法 (5.24) 将具有与 Newton 方法相一致的二次收

敛速度. 然而, 我们也面临着与 Newton 型求解算法类似的问题, 即需要在每次迭代中更新完整的雅可比矩阵, 这将导致巨大的计算开销. 为了解决这一问题, 我们提出了几种实用的预条件策略, 期望能够高效地求解所涉及到的非线性方程组:

1) **常数预条件子**, 即取 $\boldsymbol{\mathcal{M}}_k = \alpha\mathbf{I}$, 其中 α 是 $(0,1]$ 区间内的小正实数. 这种预条件策略由于无需计算非线性方程组的雅可比矩阵, 其计算成本基本可以忽略. 但是, 对于不同的问题, 往往需要手动调整参数 α 的值. 如果我们置 $\alpha = 1$, 则其退化为原始的 AA 算法. 需要注意的是, 根据误差估计 (5.29), 如果预条件子 $\boldsymbol{\mathcal{M}}_k$ 与非线性方程组的雅可比矩阵相差太远, 那么常数预条件 AA 算法只能以线性速度收敛, 甚至在某些情况下发散.
2) **全雅可比预条件子**, 即取 $\boldsymbol{\mathcal{M}}_k = jac(\boldsymbol{\mathcal{F}}_k)$. 本质上, 这种预条件策略是使用 AA 加速 Newton 方法. 不过需要注意的是它的计算成本很高.
3) **对角雅可比预条件子**, 即取 $\boldsymbol{\mathcal{M}}_k = diag(jac(\boldsymbol{\mathcal{F}}_k))$. 常数预条件子非常简单, 但不一定有效; 全雅可比预条件子在收敛速度上是最佳选择, 但计算代价非常高. 为了平衡 "好的预条件子" 和 "计算代价", 对角雅可比预条件子是一个备用选项, 而且其计算相对便宜.
4) **块对角雅可比预条件子**, 取全雅可比矩阵的块对角部分作为预条件子, 即

$$diagBlock\left(jac(\boldsymbol{\mathcal{F}})\right) = \begin{bmatrix} \partial_{\mathbf{P}^x}\boldsymbol{\mathcal{F}}^x & \mathbf{0} \\ \mathbf{0} & \partial_{\mathbf{P}^y}\boldsymbol{\mathcal{F}}^y \end{bmatrix}$$

这种预条件策略在收敛性能和计算负担之间实现了良好的平衡. 与全雅可比预条件子相比, 它降低了计算成本. 虽然它的计算成本相较于对角雅可比预条件子更高, 但它具备更优越的收敛特性.

另一个重要的观察是, 与 Newton 型求解算法和基于梯度的方法不同, AA 迭代方法中的新迭代是前几步迭代的线性组合. 这个特点带来了一个优势, 即可以避免频繁的更新计算成本高昂的预条件子. 为了提高计算效率, 我们每隔 N_{update} 步更新一次预条件子, 而非在每次迭代时都进行更新. 在本章所解决的参数化问题中, 所提出的预条件 AA 算法通常在进行预条件子更新的 1-2 步之内便能收敛.

基于以上讨论, 带有动态更新策略的预条件 Anderson 加速算法总结在算法5.1中.

5.3.2 算法收敛性分析

由第5.3.1节中的分析, 我们有如下结果:

定理 5.1. 设 $\mathbf{x}^*$ 是非线性方程 $\boldsymbol{\mathcal{F}}(\mathbf{x}) = \mathbf{0}$ 的解. 如果非线性预条件子 $\boldsymbol{\mathcal{M}}_k$ 满足

$$\left\|\mathbf{I} - \boldsymbol{\mathcal{M}}_k^{-1}\boldsymbol{\mathcal{J}}_k\right\| < 1, \tag{5.32}$$

那么预条件 Anderson 加速算法收敛. 进一步地, 我们有如式 (5.29) 的误差估计.

Algorithm 5.1: 预条件 Anderson 加速算法: **PreAA(m)**

Input: $\mathbf{P}_0$: 内部控制顶点的初始猜测;
$\boldsymbol{\mathcal{F}}(\mathbf{P})$: 非线性方程组;
m: Anderson 加速的窗口大小;
max_iter: 最大迭代次数;
tol: 残差范数容差;
N_{update}: 间隔 N_{update} 步更新预条件子.

Output: $\boldsymbol{x}$: NURBS 参数化.

for $k = 0; k \leq max_iter; ++k$ **do**
  计算残差向量 $\boldsymbol{\mathcal{F}}_k = \boldsymbol{\mathcal{F}}(\mathbf{P}_k)$; // (5.7) 或 (5.14)
  // 终止准则
  if $\|\boldsymbol{\mathcal{F}}_k\| < tol$ **then**
    break;
  end
  // 更新预条件子
  if k 能被 N_{update} 整除 **then**
    计算预条件子 $\boldsymbol{\mathcal{M}}$; // 第5.3.1节
  end
  // Anderson 加速
  计算 $m_k = \min\{m,\ k\}$;
  求解 $\boldsymbol{\mathcal{M}}\boldsymbol{\mathcal{F}}_k^{\text{pre}} = -\boldsymbol{\mathcal{F}}_k$, 以计算预条件的残差向量 $\boldsymbol{\mathcal{F}}_k^{\text{pre}}$;
  求解最小二乘问题

$$\begin{aligned} \arg\min_{\boldsymbol{\alpha}} \quad & \left\| \sum_{i=0}^{m_k} \boldsymbol{\mathcal{F}}_{k-m_k+i}^{pre} \boldsymbol{\alpha} \right\|_2^2 \\ \text{s.t.} \quad & \sum_{i=0}^{m_k} \alpha_i = 1, \end{aligned}$$

  以计算 $\boldsymbol{\alpha}^{(k)} = \left(\alpha_0^{(k)}, \alpha_1^{(k)}, \ldots, \alpha_{m_k}^{(k)}\right)^T$;
  更新 $\mathbf{P}_{k+1} = \sum_{i=0}^{m_k} \alpha_i^{(k)} \left(\mathbf{P}_{k-m_k+i} + \boldsymbol{\mathcal{F}}_{k-m_k+i}^{pre}\right)$;
end

注 5.2. 对于线性方程组情形, 即

$$\boldsymbol{\mathcal{F}}(\mathbf{x}) = \mathbf{A}\mathbf{x} - \mathbf{b} = \mathbf{0},\ \mathbf{x} \in \mathbb{R}^n, \tag{5.33}$$

分解系数矩阵 $\mathbf{A}$

$$\mathbf{A} = \mathbf{D} + \mathbf{R}, \tag{5.34}$$

其中矩阵 $\mathbf{D}$ 的所有副对角线分量和矩阵 $\mathbf{R}$ 的所有对角线分量都为零. 如果

$$\left\|\mathbf{I}-\mathbf{D}^{-1}\mathbf{A}\right\|<1, \tag{5.35}$$

那么预条件 AA 算法收敛. 这包含了 Pratapa 和 Suryanarayana 2016 年的结果[141].

近期的研究大多仅考虑使用 Anderson 加速 Picard 迭代的收敛性, 如[134,142–144]. 本研究则拓宽这一分析范畴, 将其延伸至预条件不动点迭代中 Anderson 加速的作用. 研究的主要目的在于深入理解 Anderson 加速在预条件不动点迭代中的收敛行为.

下面的定理给出了预条件 AA 算法残差的上界估计.

定理 5.2. 设 $\mathbf{x}^*$ 是非线性方程 $\boldsymbol{\mathcal{F}}(\mathbf{x})=\mathbf{0}$ 的解, 且对每一迭代 k, 非线性预条件子 $\boldsymbol{\mathcal{M}}_k$ 满足

$$\left\|\mathbf{I}-\boldsymbol{\mathcal{M}}_k^{-1}\boldsymbol{\mathcal{J}}_k\right\|\leq L_k. \tag{5.36}$$

记 $w_k=-\boldsymbol{\mathcal{M}}_k^{-1}\boldsymbol{\mathcal{F}}(\mathbf{x}_k)$, $\nabla w_k=w_{k+1}-w_k$, $\nabla x_k=x_{k+1}-x_k$,

$$\mathbf{E}_k=\left[\nabla x_{k-m},\nabla x_{k-m+1},\cdots,\nabla x_{k-1}\right], \tag{5.37}$$

且

$$\mathbf{W}_k=\left[\nabla w_{k-m},\nabla w_{k-m+1},\cdots,\nabla w_{k-1}\right]. \tag{5.38}$$

那么, 我们有如下估计

$$\left\|e_{k+1}^{AA}\right\|\leq L\bar{\Gamma}\sum_{j=0}^{m}\left\|e_{k-j}^{AA}\right\|, \tag{5.39}$$

其中

$$L=\max\{L_k,L_{k-1},\ldots,L_{k-m}\}, \tag{5.40}$$

$$\bar{\Gamma}=\max\{2\left\|\Gamma_k\right\|_\infty,\ 1+\left\|\Gamma_k\right\|_\infty\}, \tag{5.41}$$

$$\Gamma_k=\underset{\Gamma\in\mathbb{R}^m}{\arg\min}\left\|\mathbf{w}_k-\mathbf{W}_k\Gamma\right\|_2^2. \tag{5.42}$$

证明. 根据式 (5.24) 中的更新规则, 我们有第 $k+1$ 步的预条件 Anderson 加速解

$$\begin{aligned}\mathbf{x}_{k+1}^{AA}&=\mathbf{x}_k^{AA}+\mathbf{w}_k-\sum_{j=1}^{m}\left[(\mathbf{x}_{k-m+j}^{AA}-\mathbf{x}_{k-m+j-1}^{AA})+(\mathbf{w}_{k-m+j}-\mathbf{w}_{k-m+j-1})\right]\Gamma_{k,j}\\&=(\mathbf{x}_k^{AA}+\mathbf{w}_k)(1-\Gamma_{k,m})+\sum_{j=1}^{m-1}(\mathbf{x}_{k-m+j}^{AA}+\mathbf{w}_{k-m+j})(\nabla_j\Gamma_{k,j})+(\mathbf{x}_{k-m}^{AA}+\mathbf{w}_{k-m})\Gamma_{k,1},\end{aligned} \tag{5.43}$$

其中 $\nabla_j \Gamma_{k,j} = \Gamma_{k,j+1} - \Gamma_{k,j}$.

由于

$$\begin{aligned}\mathbf{x}^* &= \left(1 + \sum_{j=1}^{m} (\Gamma_{k,i} - \Gamma_{k,i})\right) \mathbf{x}^* \\ &= [1 - \Gamma_{k,m}]\mathbf{x}^* + \sum_{j=1}^{m-1} (\nabla_j \Gamma_{k,j})\, \mathbf{x}^* + \Gamma_{k,1}\mathbf{x}^*,\end{aligned} \tag{5.44}$$

我们有

$$\begin{aligned}\mathbf{e}_{k+1}^{AA} &= \mathbf{x}_{k+1}^{AA} - \mathbf{x}^* \\ &= (\mathbf{x}_k^{AA} + \mathbf{w}_k - \mathbf{x}^*)(1 - \Gamma_{k,m}) + \sum_{j=1}^{m-1} (\mathbf{x}_{k-m+j}^{AA} + \mathbf{w}_{k-m+j} - \mathbf{x}^*)(\nabla_j \Gamma_{k,j}) \\ &\quad + \ (\mathbf{x}_{k-m}^{AA} + \mathbf{w}_{k-m} - \mathbf{x}^*)\Gamma_{k,1} \\ &= \mathbf{e}_k^{AA}(1 - \Gamma_{k,m}) + \sum_{j=1}^{m-1} \mathbf{e}_{k-m+j}^{AA}(\gamma_{j+1} - \gamma_j) + \mathbf{e}_{k-m}^{AA}\Gamma_{k,1}.\end{aligned} \tag{5.45}$$

将式 (5.29) 代入上式中得到

$$\left\| e_{k+1}^{AA} \right\| \le L\bar{\Gamma} \sum_{j=0}^{m} \left\| e_{k-j}^{AA} \right\|. \tag{5.46}$$

□

注 5.3. 如果选择预条件子 $\boldsymbol{\mathcal{M}}_k$ 充分接近原非线性方程组的雅可比矩阵 $\boldsymbol{\mathcal{J}}_k$, 即有下式

$$\left\| \mathbf{I} - \boldsymbol{\mathcal{M}}_k^{-1} \boldsymbol{\mathcal{J}}_k \right\| \le L_k \le \frac{1}{(m+1)\bar{\Gamma}} < 1$$

成立, 那么由式 (5.39) 可知, 预条件 Anderson 加速算法收敛.

5.4 数值实验与对比分析

在本节中, 我们将对本章所提出的参数化方法进行数值研究, 并检验所提出的预条件 AA 算法在这一问题中的有效性. 同时还将与前两章中的方法[127,132]进行比较.

5.4.1 算法实现细节

本方法采用 C++ 进行实现. 所有数值实验均在一台笔记本电脑上进行 (MacBook Pro 14-inch 2021, 配备 Apple M1 Pro CPU 和 16GB RAM). 该实现基于开源 C++ 库 G+Smo(Geometry + Simulation Modules)[145,146]. 线性代数中的矩阵/向量操作和求解相

关线性方程组则使用了 C++ 模板库 Eigen[147].

算法参数设置: 算法5.1中涉及几个参数. 我们将所有参数设置为默认值, 但通过调整这些参数仍可进一步提高收敛速度. 我们将窗口大小设定为 $m = 5$, 并设置 $N_{update} = 10$, 即每 10 次迭代更新一次预条件子. 至于算法终止准则, 设定最大迭代次数 $max_iter = 1000$ 和残差范数的容差 $tol = 1e-5$.

5.4.2 不同非线性求解算法的性能比较

在本节中, 我们将要探讨采用不同预条件策略的预条件 AA 算法的数值性能. 同时, 我们还将其与其他求解算法进行比较, 包括标准 Newton 方法, 标准 AA 算法 (AA(m)) 以及最近开发的两种方法[148]: AA 复合 AA 算法 (AA(m,AA(m))) 和 AA 自适应复合 AA 算法 (AAadp(m,AA(m))).

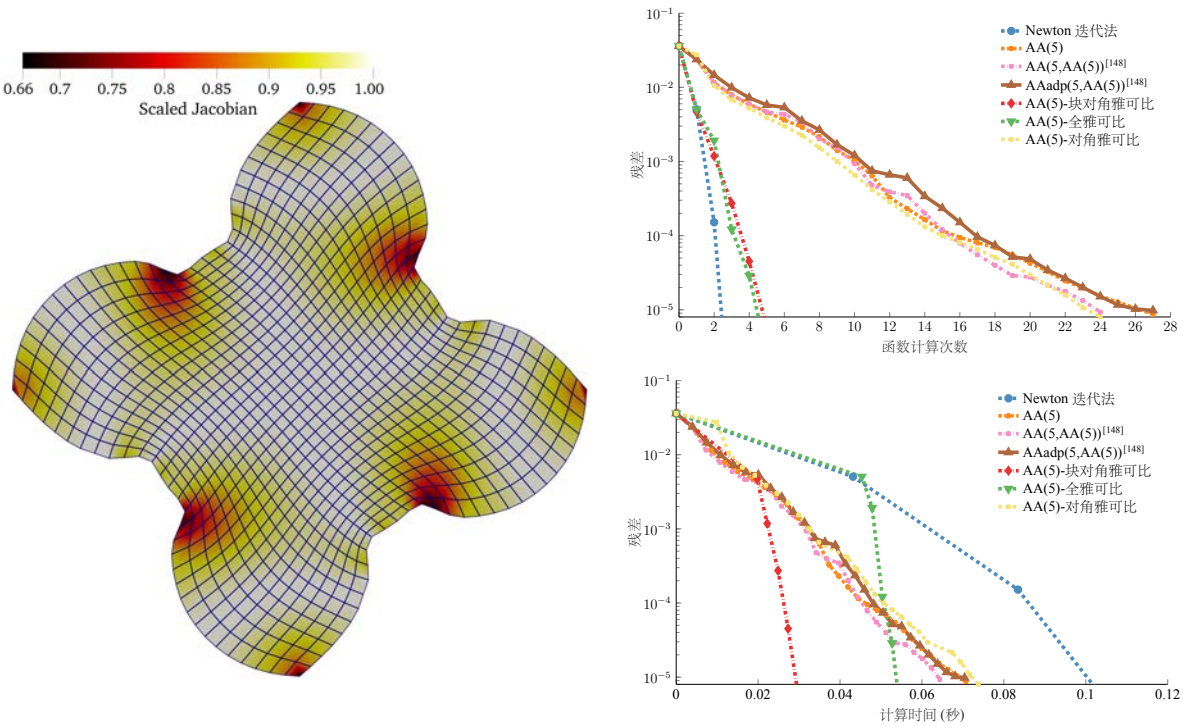


图 5.3 阳转子: 不同非线性求解算法的数值表现对比

Fig. 5.3 Male Rotor: Numerical Performance Comparison of Different Nonlinear Solvers

图 5.3 展示了双螺杆压缩机中阳转子的固体部分计算域. 我们从其由四条 B 样条曲线组成的边界表示出发, 利用 Spring 模型方法[52] 来构造初始参数化. 右边的两幅图分别展示了不同求解算法的残差范数随着函数计算次数和计算时间的变化曲线. 对比之下, Newton 方法需要最少的函数计算次数, 可以在 4 次迭代内收敛. 在这个例子中, AA(5), AA(5,AA(5)), AAadp(5,AA(5)) 和带有对角雅可比矩阵预条件子的预条件 AA 算法在 25 次迭代后收敛. 而带有全雅可比矩阵预条件子的预条件 AA 算法和带有块

对角雅可比矩阵预条件子的预条件 AA 算法分别在经过 9 次和 10 次迭代后收敛, 虽然这比标准 Newton 方法要差, 但比其他 AA 算法要好.

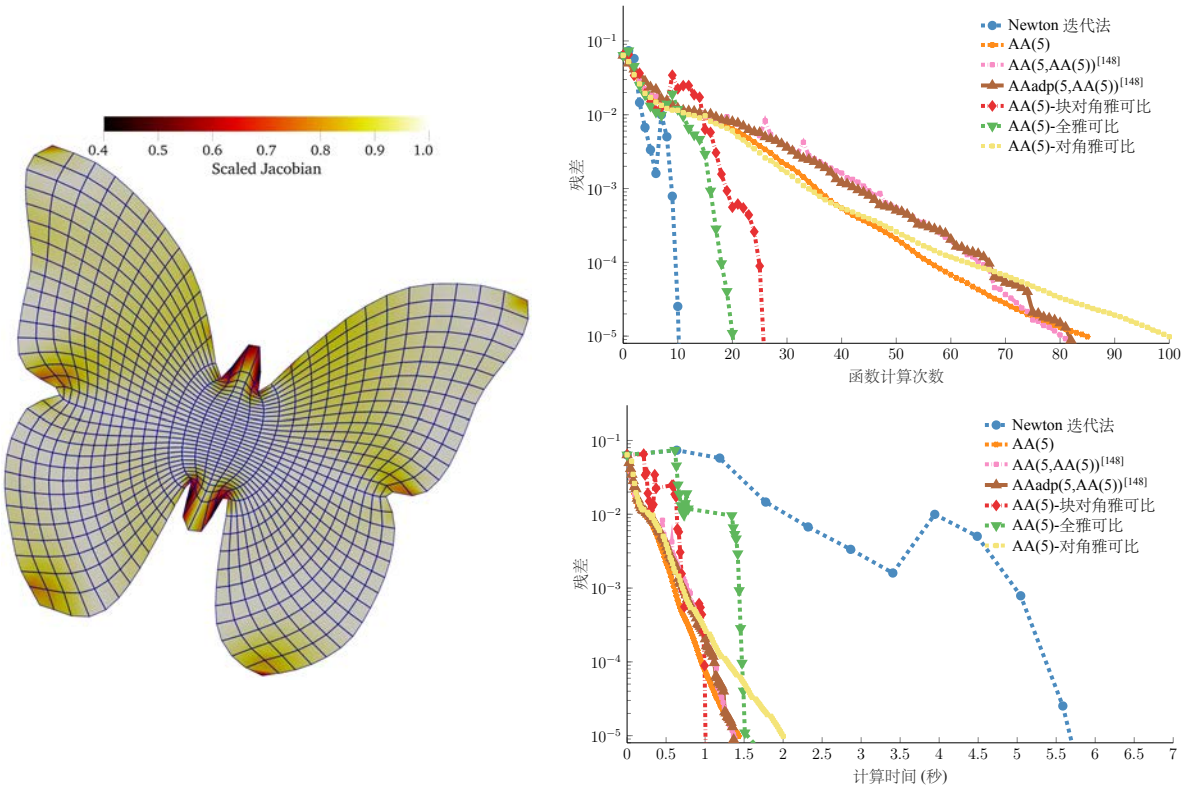


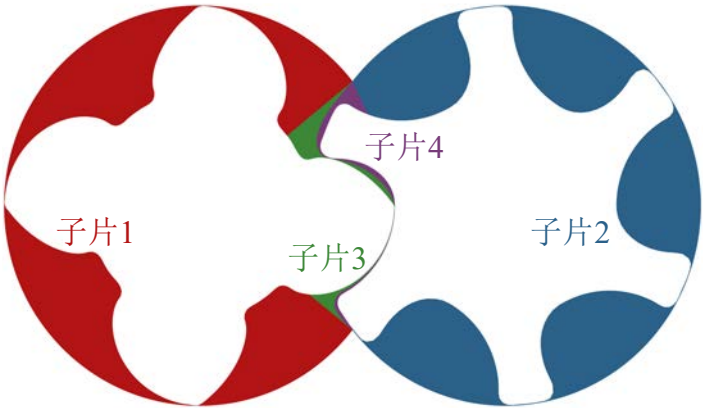
图 5.4　蝴蝶: 不同非线性求解算法的数值表现对比

Fig. 5.4　Butterfly: Numerical Performance Comparison of Different Nonlinear Solvers

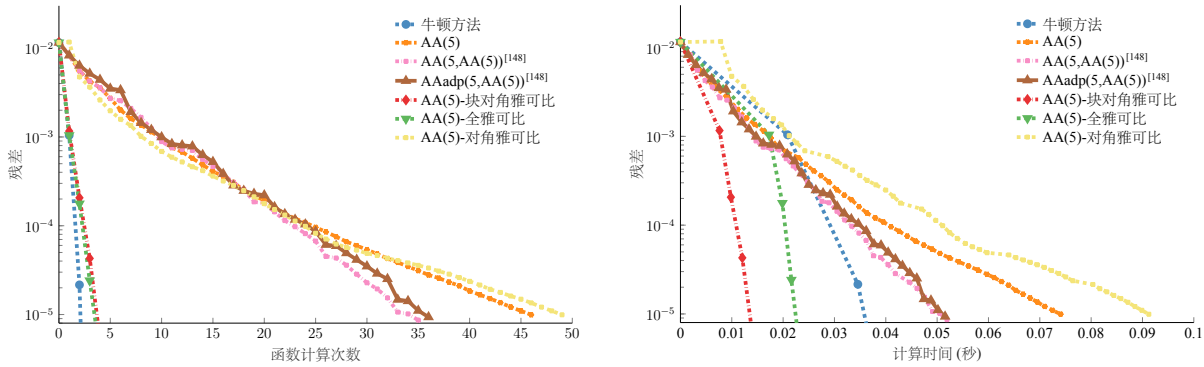
在这个例子中, AA 算法及其变种在计算时间方面都优于标准 Newton 方法. 尽管第 5.2.2 节和第 5.2.4 节中所推导出的解析雅可比公式极大的提升了计算效率, 但在标准的 Newton 方法中需要在每次迭代中更新完整的雅可比矩阵, 计算代价高. 在本例中, 块对角雅可比矩阵预条件子的预条件 AA 算法比标准 Newton 方法快四倍以上.

如图 5.4 所示, 蝴蝶形状的几何模型也展现出类似的数值表现. 在这个例子中, 我们基于块对角雅可比矩阵预条件的 AA 算法比标准 Newton 方法快近六倍. 这里需要说明的是, 在我们的预条件 AA 算法的收敛曲线中出现了振荡. 这是因为为了进行公平比较, 我们记录了原始 (无预条件) 残差范数, 而预条件残差范数仍然稳定下降.

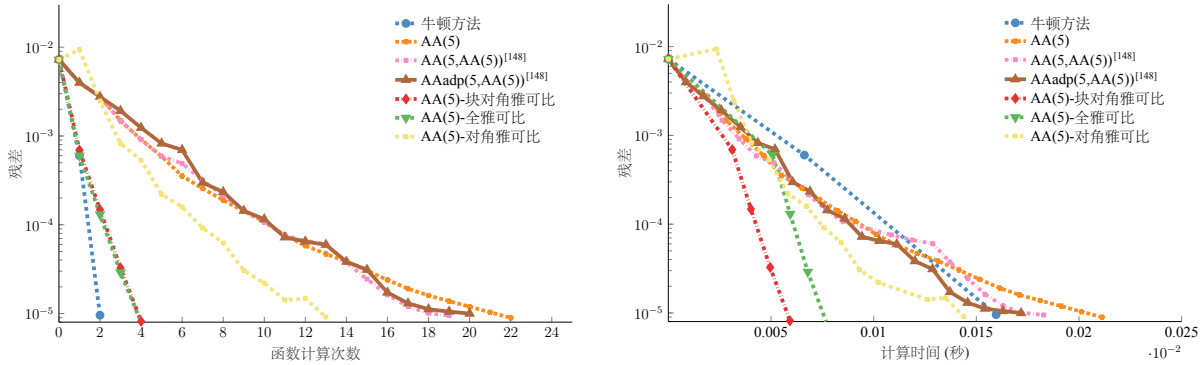
接下来, 我们在多片参数化上进行测试. 图 5.5(a) 展示了由 4 片 B 样条参数化的旋转双螺杆压缩机的一个切片. 不同非线性求解算法在两个具有代表性的子片上 (即子片 1 和子片 4) 的收敛表现分别如图 5.5(b) 和图 5.5(c) 所示. 本章提出的带有块对角线雅可比预条件子的预条件 AA 算法在计算时间方面表现最佳. 另外两个子片上的数值性能具有类似的表现.



(a) 一个由四片参数化的双螺杆几何



(b) 子片 1: 不同非线性求解算法的数值表现



(c) 子片 4: 不同非线性求解算法的数值表现

图 5.5　不同非线性求解算法在四片参数化的双螺杆几何上的表现

Fig. 5.5　Performance of Different Nonlinear Solvers on Four-Patch Twin-screw Geometry

5.4.3　缩放 H^1 方法对参数化质量的影响

如第 5.2.3 节所述, 通过求解原始的非线性方程组(5.7)在 H^2 空间中的离散方程生成的参数化结果可能具有不均匀的单元. 具体来说, 所生成的网格线会被边界附近的凹区域吸引而被凸区域排斥, 如图 5.2 所示. 因此, 我们在第 5.2.3 节中介绍了缩放的非线性方程组(5.7)在 H^1 空间中的离散. 在本节中, 我们对比两种方法所生成的参数化.

图 5.6 展示了一个典型的例子. 左图展示了通过求解非线性方程组 (5.7) 得到的参

数化结果, 在红色圆圈内有自交点. 表 5.1 中数据表明该模型的雅可比行列式最小值为负, 这也进一步印证了这一点. 这说明, 尽管在无穷维空间中, 调和映射的双射性由 Radó-Kneser-Choquet 定理保证, 但在有限维的样条空间中, 它仍然有可能产生无效的参数化结果. 文献[72]提出可以通过多次细化以对调和映射有更好的逼近. 然而, 问题是细化会带来更多的控制顶点, 使模型的规模变大. 利用本章提出的缩放的 H^1 离散得到的参数化结果如图5.6中的右图所示, 它的参数化质量得到了极大的提升. 尤为重要的是, 这里并没有增加控制顶点的个数, 因此在实际应用中更受欢迎. 此外, 表 5.1 统计了不同模型的参数化质量指标, 其中每个模型的第三行和第四行分别展示了 H^2 离散和 H^1 离散的结果. 可以看出, 本章提出的缩放的 H^1 离散大大提高了参数化质量.

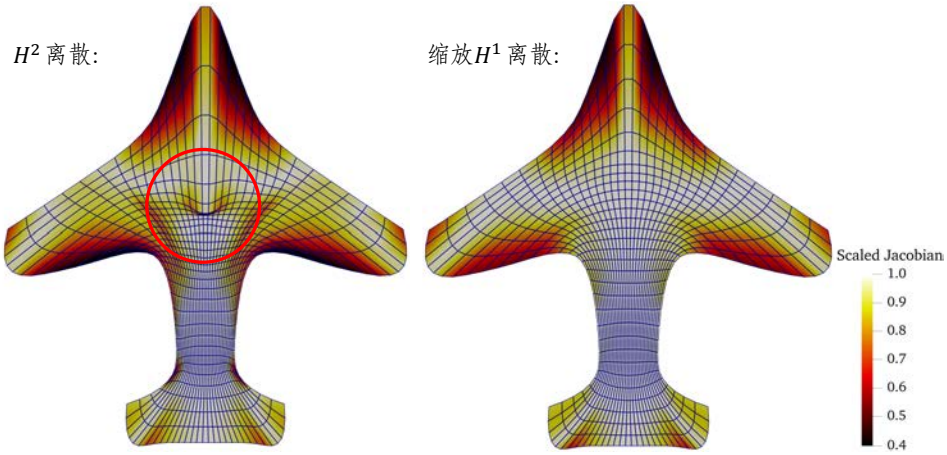


图 5.6　飞机: H^2 离散和缩放的 H^1 离散生成的参数化结果
Fig. 5.6　Plane: Parameterization Results by H^2 and Scaled H^1 Discretization

5.4.4　与现有方法的对比分析

本节中, 我们将所提出的方法与现有的参数化方法进行对比. 具体将关注两种方法: 一种是第 3 章中基于障碍函数的方法[127], 另一种是第 4 章中基于惩罚函数的方法[132]. 这两种方法都试图通过最小化 Winslow 函数来近似调和映射. 这两种方法的主要区别在于, 基于惩罚函数的方法通过引入惩罚项来避免额外的消除折叠步. 然而, 由于这两种方法本质上都是最小化 Winslow 泛函, 它们在收敛后的参数化结果应该是一致的. 具体来说, 这两种方法求解以下非线性优化问题:

$$\underset{\mathbf{P}_i, i\in\mathcal{I}_I}{\arg\min} \quad \int_{\hat{\Omega}} \left(\mathscr{E}^{\text{Winslow}} + \lambda \mathscr{E}^{\text{uniform}}\right) d\hat{\Omega}, \tag{5.47}$$

其中 $\mathscr{E}^{\text{Winslow}}$ 是式 (5.3) 中的 Winslow 泛函. 为展示不同离散化方法之间的差异, 在数值实验中将平衡参数 λ 设置为 0. 为了公平比较, 我们在相同的计算环境下使用 G+Smo 框架重新实现了这两种方法, 这比前两章所采用的原始 MATLAB 实现要高效得多. 有

关这两种方法与现有方法的对比, 请参阅本论文第 3 章和第 4 章或文献[127,132].

表 5.1　不同方法所构造参数化结果的量化统计数据
Tab. 5.1　Quantitative Data for Parameterization Results Obtained by Different Methods

模型	自由度	参数化方法	比例雅可比		均匀性		时间 (秒)
			最小值	平均值	最大值	平均值	
鸭子 (图 5.2)	160	障碍函数[127]	0.2986	0.9243	62.9882	0.5913	0.0393
		惩罚函数[132]	0.2986	0.9243	62.9875	0.5912	0.0722
		H^2 离散 (5.7)	0.2444	**0.9276**	131.3254	0.7845	**0.0066**
		H^1 离散 (5.14)	**0.3022**	0.9257	**27.0234**	**0.3545**	0.0195
阳转子 (图 5.3)	1352	障碍函数[127]	0.3737	**0.9713**	2.7579	0.1444	0.2204
		惩罚函数[132]	0.3738	**0.9713**	2.7611	0.1444	0.2984
		H^2 离散 (5.7)	0.3140	**0.9713**	2.7749	0.1451	**0.0590**
		H^1 离散 (5.14)	**0.5293**	0.9644	**0.7342**	**0.0688**	0.1544
蝴蝶 (图 5.4)	4418	障碍函数[127]	0.3095	0.9414	77.1353	0.6885	16.0262
		惩罚函数[132]	**0.3096**	0.9414	77.1177	0.6886	27.4425
		H^2 离散 (5.7)	0.2913	0.9416	114.4743	0.7014	**1.5317**
		H^1 离散 (5.14)	0.2319	**0.9596**	**30.2282**	**0.3838**	4.5948
飞机 (图 5.6)	144	障碍函数[127]	0.3575	0.9145	45.6981	1.0526	**0.0385**
		惩罚函数[132]	0.3575	0.9145	45.6983	1.0526	0.0539
		H^2 离散 (5.7)	-1	-	-	-	-
		H^1 离散 (5.14)	**0.3716**	**0.9379**	**28.6562**	**0.7400**	0.0538
阳转子机匣 (图 5.7(左))	1616	障碍函数[127]	-1	-	-	-	-
		惩罚函数[132]	-1	-	-	-	-
		H^2 离散 (5.7)	0.3469	0.9407	36.5805	4.6449	**0.0436**
		H^1 离散 (5.14)	**0.4611**	**0.9474**	**26.1502**	**4.5278**	0.0531
阴转子机匣 (图 5.7(右))	1616	障碍函数[127]	-1	-	-	-	-
		惩罚函数[132]	-1	-	-	-	-
		H^2 离散 (5.7)	0.2041	0.9443	60.6725	4.9954	**0.0344**
		H^1 离散 (5.14)	**0.4388**	**0.9539**	**33.6031**	**4.7157**	0.1320

在表 5.1 中, 我们比较了不同参数化方法在质量指标和计算时间方面的表现. 正如我们所预期的, 基于障碍函数的方法[127] 和基于惩罚函数的方法[132] 在所有模型中生成的参数化结果几乎相同, 这是因为它们都致力于最小化 Winslow 函数. 我们在 Sobolev 空间 H^1 中的离散化产生了更高质量的结果, 不仅在正交性上有所改善, 而且在均匀性方面也有所提升. 在计算效率方面, 我们的方法比其他方法要快得多 (除了飞机模型). 值得注意的是, 对于使用有限内存 BFGS 求解算法的基于障碍函数和基于惩罚函数的方法, 在具有 4418 个自由度的蝴蝶模型中收敛缓慢. 然而, 本章方法的计算效率比这两种方法提高了近 4 倍.

如图 5.7 所示, 所产生的平面参数化可应用于某些工业应用中的 $2.5D$ 体参数化.

为了构造这些体参数化, 我们首先为单个切片构造适合分析的平面参数化, 接着对得到的平面参数化结果进行旋转, 并不断地沿着 z 轴方向扫掠. 这样我们得到的参数化结果在第三个参数方向上次数为 1. 对于这些具有极端纵横比的几何形状, 极小化修正的 Winslow 泛函的方法[127,132] 遇到了数值困难, 这一现象也可以在图 5.1 中观察到. 如表 5.1 所示, 缩放雅可比矩阵的最小值为负数, 这意味着这两种方法所得到的参数化是无效的. 而在 H^2 和 H^1 空间的离散化中, 我们都可以获得有效的参数化结果. 从图 5.7 中可以看出, H^2 离散产生了更高质量的结果. 缺点是相较于 H^2 离散, 缩放的 H^1 离散需要更多的计算时间. 然而, 利用我们提出的块对角雅可比预条件 AA 算法, 其计算时间仍然是可以接受的.

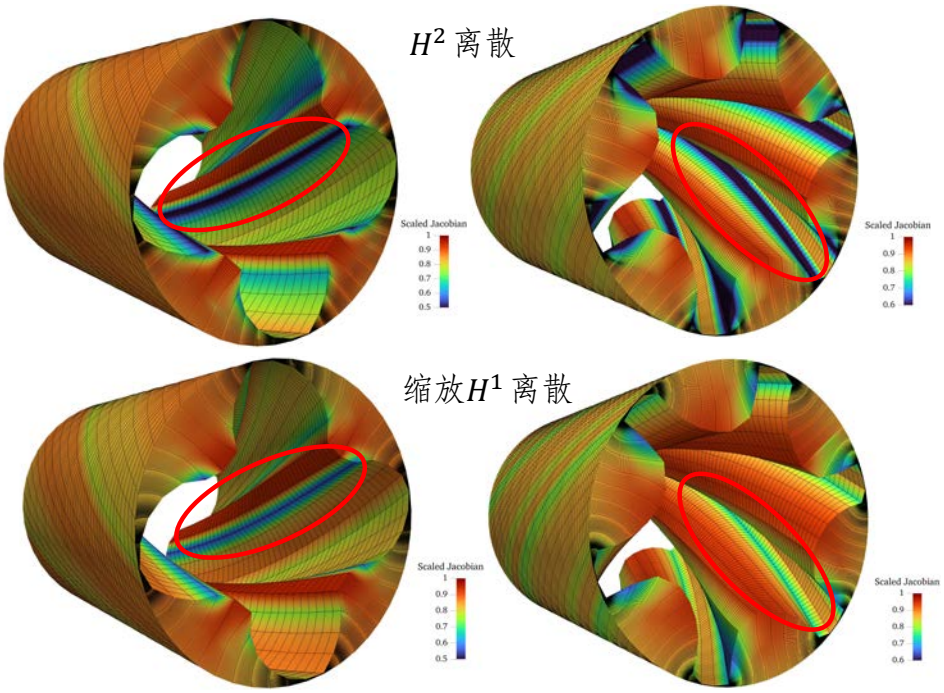


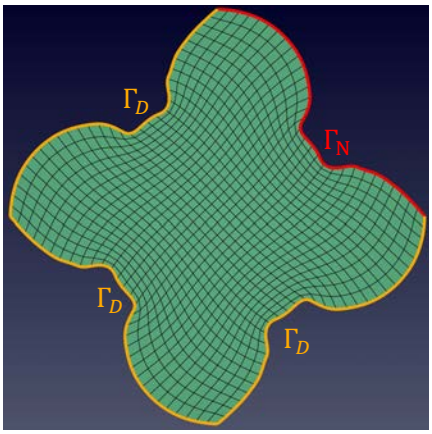
图 5.7　螺杆与机匣: H^2 离散和缩放的 H^1 离散生成的参数化结果
Fig. 5.7　Screw and Casing: Parameterization Results from H^2 and Scaled H^1 Discretization

5.5　在等几何分析中的应用

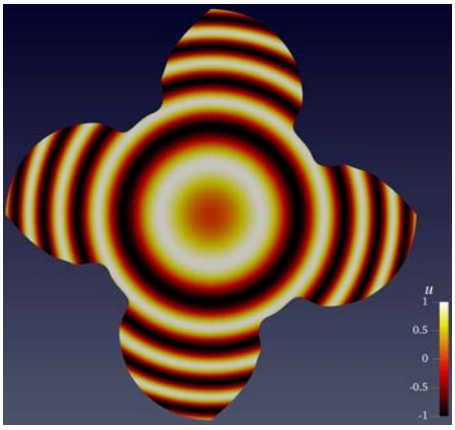
为了证明本章生成的参数化结果适用于等几何分析, 我们考虑以下带有混合边界条件的 Poisson 问题:

$$\begin{cases} -\Delta \mathscr{U} = f, & \boldsymbol{x} \in \Omega, \\ \mathscr{U} = g, & \boldsymbol{x} \in \Gamma_D, \\ \nabla \mathscr{U} \cdot \mathbf{n} = h, & \boldsymbol{x} \in \Gamma_N, \end{cases} \tag{5.48}$$

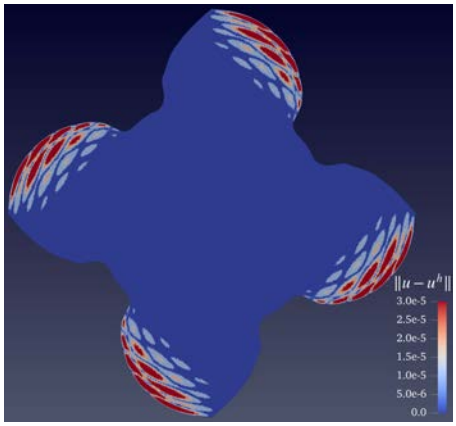
其中 $f \in L^2(\Omega) : \Omega \to \mathbb{R}$ 是给定的源项, $\mathbf{n}$ 是边界的单位外法线向量, Γ_D 和 Γ_N 分别是指定 Dirichlet 和 Neumann 边界条件的边界, 且满足 $\bar{\Gamma}_D \cup \bar{\Gamma}_N = \partial\Omega$, $\Gamma_D \cap \Gamma_N = \varnothing$.



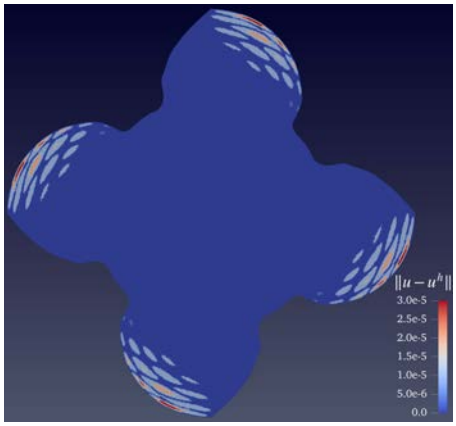
(a) 边界条件



(b) 精确解



(c) H^2 离散参数化结果的绝对误差分布



(d) H^1 离散参数化结果的绝对误差分布

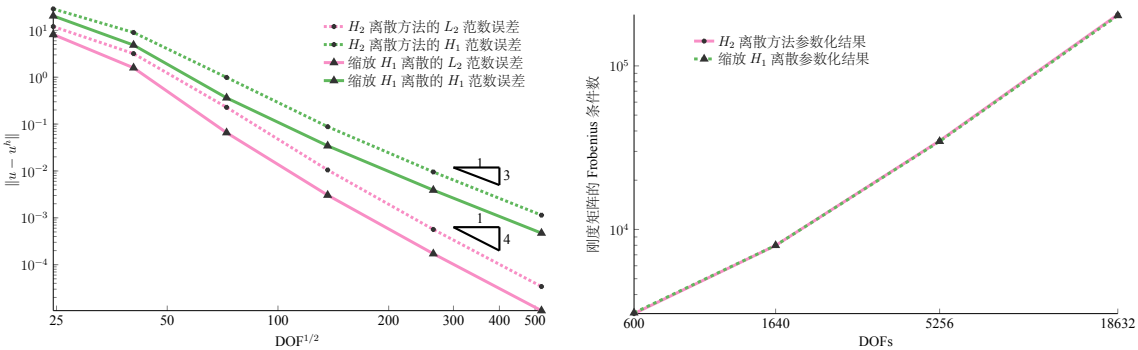
图 5.8　平面阳转子几何: 边界条件、精确解及绝对误差分布

Fig. 5.8　Planar Male Rotor: Boundary Conditions, Exact Solution, and Absolute Error Distribution

首先, 我们考虑如图5.8(a)所示的阳转子上带有混合边界条件的 Poisson 方程 (5.48). 在这个实例中, Dirichlet 边界条件通过 L_2 投影方法施加. 其精确解为 $\mathscr{U}^{\text{exact}} = \sin(\frac{x^2+y^2}{50})$, 如图 5.8(b) 所示. 源项为 $f = -\Delta\mathscr{U}^{\text{exact}}$. 如图 5.8(c) 和图 5.8(d) 所示, 基于 H^2 离散和 H^1 离散生成的参数化结果求解该方程得到的误差分布具有显著差异. 在这种情况下, 由 H^1 离散构造的参数化比由 H^2 离散构造的参数化具有更小的误差.

图 5.9(a) 展示了 h 细化过程中误差收敛曲线. 可以看到, L^2 范数误差和 H^1 范数误差都达到了理论上的最优收敛率. 图 5.9(b) 展示了 h 细化过程中刚度矩阵 Frobenius 条件数的变化曲线. 可以看到, 随着自由度的增加, 两种参数化方法的刚度矩阵条件数均呈稳定的线性增长, 且保持在一个值较小的水平下 (自由度为 18632 时大约为 10^5). 这与文献[91]中的观察一致. 以上讨论表明, 本章方法生成的参数化适合于等几何分析.

接下来, 我们在图 5.10 所示的三维螺杆几何体上求解 Poisson 方程 (5.48). 在进行等几何分析求解之前, 我们沿着 z-轴方向对该参数化进行一次升阶, 以获得每个方向均是二次的 B 样条表示. 其精确解为 $\mathscr{U}^{\text{exact}} = e^{\frac{x^2+y^2}{r^2}}\sin(\frac{z\pi}{m})$, 其中 $r = 36$, $m = 200$.



(a) 误差收敛曲线　　(b) 刚度矩阵的 Frobenius 条件数变化曲线

图 5.9　h 细化过程中误差收敛和刚度矩阵 Frobenius 条件数变化曲线

Fig. 5.9　Error Convergence and Frobenius Condition Number of Stiffness Matrix during h-Refinement

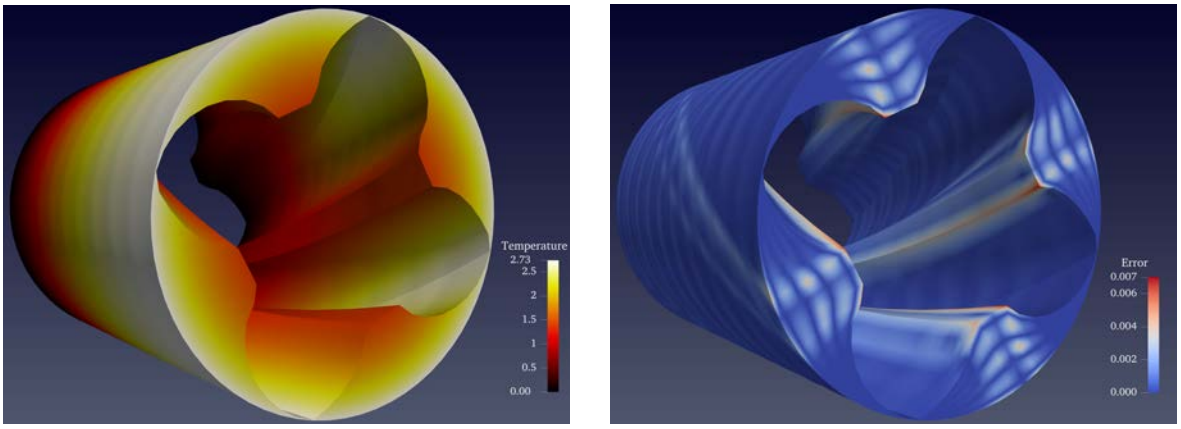


图 5.10　三维转子几何: 等几何分析数值解和绝对误差分布 (自由度为 30, 704)

Fig. 5.10　3D Rotor Geometry: IGA Solution and Absolute Error Distribution (DOFs = 30, 704).

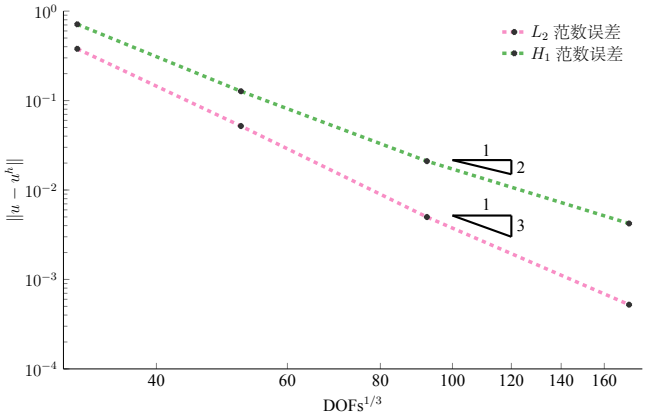


图 5.11　h 细化过程中误差收敛曲线

Fig. 5.11　Error Convergence Curve During h-Refinement

求得的等几何分析数值解及其对应的绝对误差分布如图 5.10 所示. 图 5.11 绘制了 h 细化过程中误差收敛曲线. 可以看出, L^2 范数误差和 H^1 范数误差均达到了理论

上最优的收敛速率. 这进一步说明了我们的参数化结果适用于等几何分析.

5.6 各向同性参数化方法比较

本节将全面对比前几章中提到的各向同性参数化方法, 这包括第 3 章中介绍的障碍函数方法、第 4 章中所述的惩罚函数方法、以及本章所探讨的两种椭圆参数化方法.

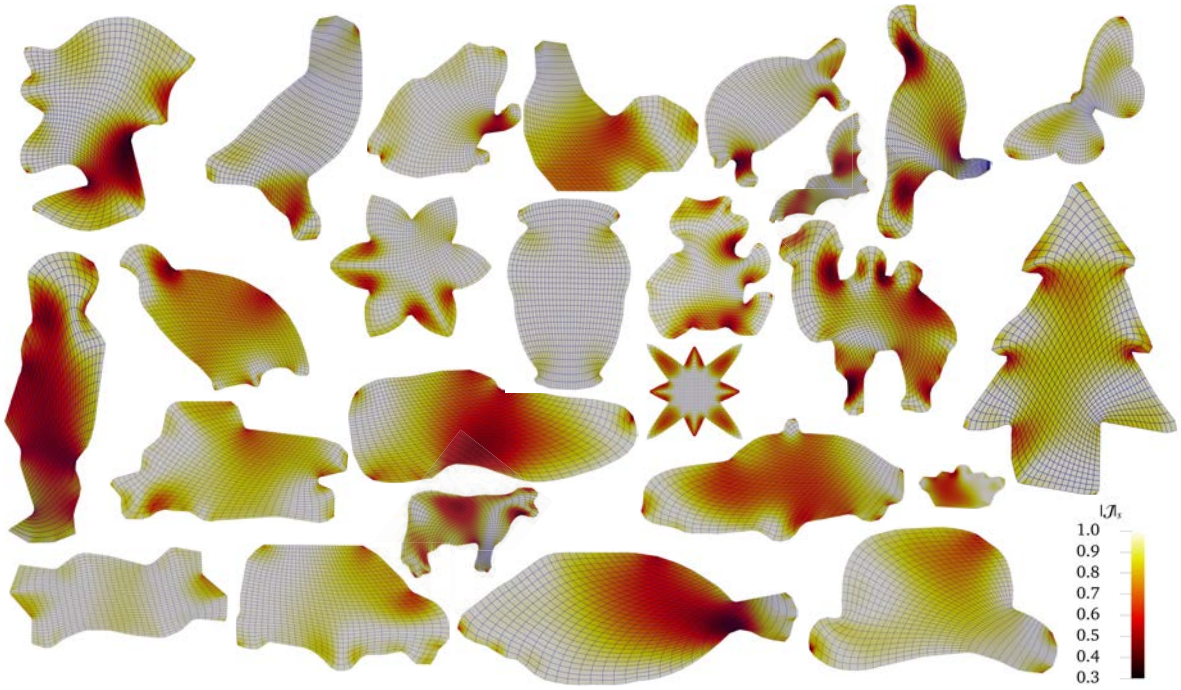


图 5.12 977 平面数据集中部分模型
Fig. 5.12 Partial Models in the 977 Planar Dataset

在大量模型上对这些参数化方法进行广泛评估, 旨在全面地了解各种参数化方法的优缺点, 并找出各方法潜在的局限性. 为此, 我们在一个包含 977 个平面模型的数据集上进行实验, 其中一部分模型如图 5.12所示. 这些模型在每个参数方向上有 25 个控制顶点, 因此由具有 625 个控制顶点的单片 B 样条曲面表示.

实验中所得到的所有结果和数据, 都是基于我们在 G+Smo 中的开源代码所获得的, 该代码的地址为: `https://github.com/gismo/gismo`. 此外, 所有的实验工作都是在 MacBook Pro 14-inch 2021 笔记本电脑上完成的.

图 5.13 展示了各种不同方法得到的参数化结果的最差质量指标, 其中包括比例雅可比的最小值 $\min\left(|\mathcal{J}|_s\right)$ 和均匀性度量的最大值 $\max\left(m^{\text{unif.}}\right)$. 值得关注的一点是, $\min\left(|\mathcal{J}|_s\right)$ 的值为负表明参数化结果中存在自相交的情况. 从图 5.13 中我们可以清晰地观察到, 在这个平面模型数据集上, 基于优化的参数化方法相对于基于椭圆参数化的方法表现出了更高的鲁棒性。并且, 我们所提出的缩放 H^1 方法相比于原始的 H^2 方法展现出了更高的成功率.

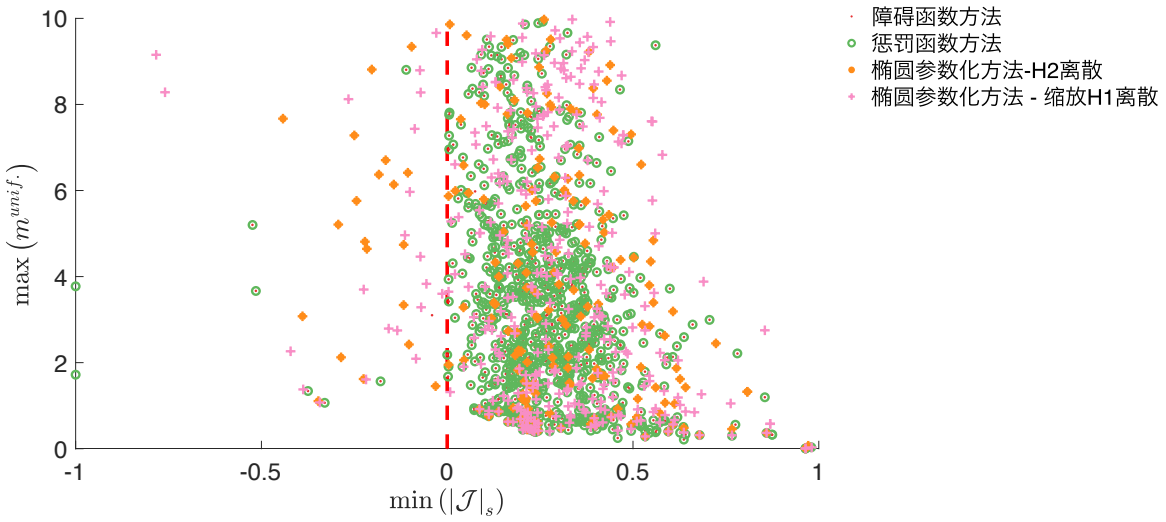


图 5.13　不同方法的最差质量指标: min $(|\mathcal{J}|_s)$ vs. max $(unif.)$

Fig. 5.13　Worst-Case Quality Metrics of Different Methods: min $(|\mathcal{J}|_s)$ vs. max $(unif.)$.

表 5.2 列出了各种参数化方法的成功率, 以供读者更全面地了解和比较这些方法的效果和性能. 图 5.14 展示了比例雅可比 $|\mathcal{J}|_s$ 和均匀性指标 $m^{\text{unif.}}$ 的平均值. 可以观察到, 基于障碍函数的方法和基于惩罚函数的方法在大多数模型中都显示出了相似的收敛结果. 这主要是因为这两种方法在优化过程中是在最小化相同的目标函数, 它们之间的区别仅在于处理双射约束的方法有所不同. 与 H^2 离散的椭圆参数化方法相比, 我们所提出的缩放 H^1 离散方法在实验中显著地提高了参数化结果的均匀性, 表明了该方法在提高参数化质量方面的优越性能.

表 5.2　不同参数化方法的成功率

Tab. 5.2　Success Rates of Different Parameterization Approaches

参数化方法	成功率
障碍函数方法	$\frac{961}{977} \simeq 98.36\%$
惩罚函数方法	$\frac{956}{977} \simeq 97.85\%$
椭圆参数化方法-H^2 离散	$\frac{608}{977} \simeq 62.23\%$
椭圆参数化方法-H^1 离散	$\frac{721}{977} \simeq 73.80\%$

图 5.15 展示了不同参数化方法在计算时间方面的表现. 可以观察到, 基于 PDE 的 H^2 离散化方法在计算时间上展现出了最佳的性能. 基于 PDE 的椭圆参数化方法的平均计算时间为 0.2 秒, 这一时间远远低于前两章中基于优化的参数化方法的平均计算时间 (约为 2 秒). 总的来说, 本章中介绍的椭圆参数化方法相较于前两章中介绍的基于优化的参数化方法表现出了更快的计算速度. 因此, 在需要考虑计算效率的应用场景中, 椭圆参数化方法是一个更为理想的选择.

最后, 正如本章一开始就强调的, 针对那些具有极端纵横比的狭长计算域, 我们仍

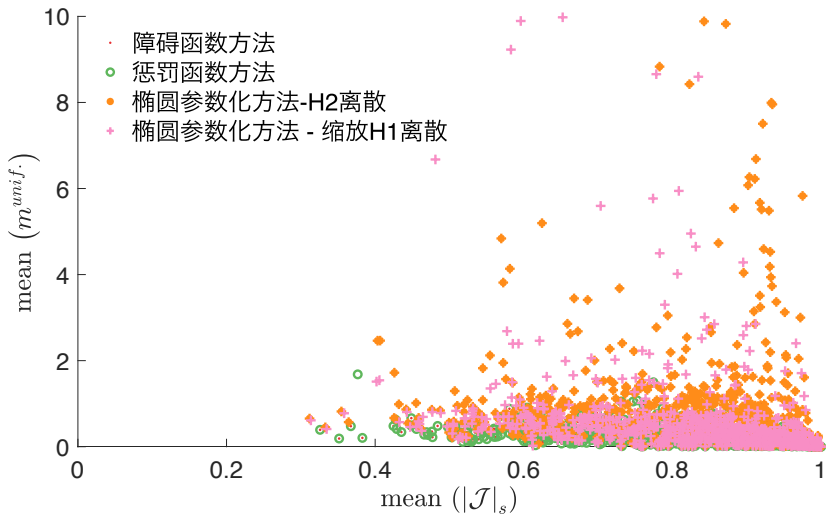


图 5.14　不同方法的平均质量指标: min $(|\mathcal{J}|_s)$ vs. max $(unif.)$

Fig. 5.14　Mean Quality Metrics of Different Methods: mean $(|\mathcal{J}|_s)$ vs. mean $(unif.)$

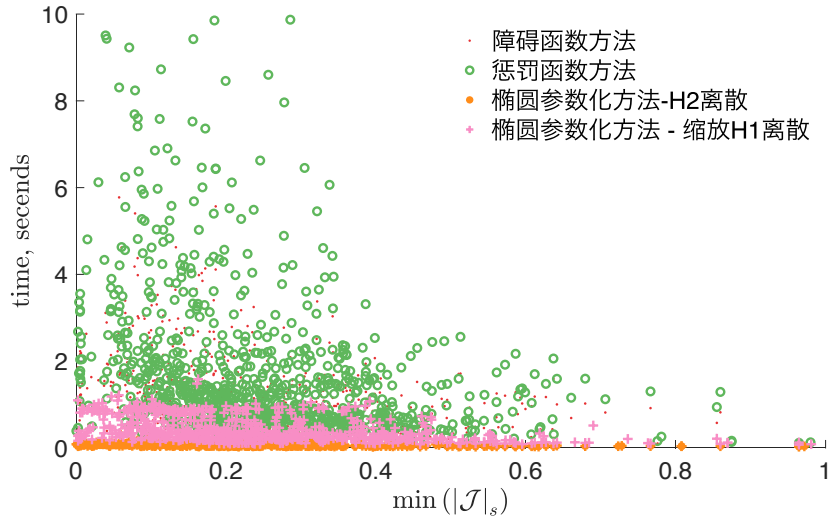


图 5.15　不同方法的计算时间

Fig. 5.15　Computation Time of Different Methods

然强烈推荐使用本章所介绍的椭圆参数化方法. 需要特别指出的是, 在当前的数据集中, 这种特殊的计算区域相对较少. 对于一般计算域的参数化问题, 我们建议的策略是先尝试使用本章所介绍的椭圆参数化方法, 然后对得到的参数化结果进行质量检验. 如果参数化结果未能达到预期, 那么可以考虑转用前面两章所介绍的基于优化的参数化方法. 这样的分阶段策略不仅灵活, 而且能更好地应对各种实际需求.

5.7　本章小结

在本章中, 我们提出了一种预条件 Anderson 加速算法框架, 目的是优化基于 PDE 的椭圆参数化技术, 以提升其数值稳定性和计算效率. 为了解决椭圆参数化在曲率变

化较大的复杂边界区域附近可能出现的单元不均匀问题, 引入了缩放函数 $\mathbb{A}$ 并给出了其在 Sobolev 空间 H^1 中的离散格式. 通过仔细推导, 给出了涉及的非线性方程组的解析雅可比矩阵, 显著提高了计算效率. 数值实验结果表明, 这一预条件 AA 框架能够显著降低每次迭代的计算负担和整体运算时间. 相较于现有方法, 本章提出的方法在处理极端纵横比的几何计算区域参数化方面展示出更为卓越的性能.

值得一提的是, 本章所引入的预条件 AA 算法具有广泛的应用潜力, 可推广至其他非线性方程组求解场景. 虽然在本章所讨论的参数化问题中, 我们成功地推导出了相关非线性方程组的解析雅可比矩阵, 但在一般情况下, 获得解析雅可比矩阵可能相当困难, 甚至不可行. 因此, 进一步探索不依赖于雅可比矩阵的预条件策略将具有重要的理论和实用价值.

为了清晰地展示本章方法的实际应用价值, 在下一章 (第6章) 中, 我们将本章所提出的参数化方法应用到真实工业场景中双螺杆压缩机械的网格生成问题中.

6 基于样条参数化的双螺杆压缩机网格生成方法

6.1 引言

容积式旋转机械在工业领域中有着十分广泛的应用, 主要用于压缩或输送各种类型的流体, 包括液体和气体. 如图 6.1 所示的双螺杆压缩机是这一类机械的一个典型代表. 它主要由两个紧密螺旋状嵌合的转子组成, 其基本工作原理是通过减少内部容积来压缩所吸入的气体, 并以更高的压力将其排出. 由于其高效且稳定的压缩性能, 双螺杆压缩机在如空气压缩和制冷系统等工业生产领域得到了广泛的应用. 因此, 即便是微小的性能提升也能在全球范围内带来显著的能源节约和碳排放降低[111].

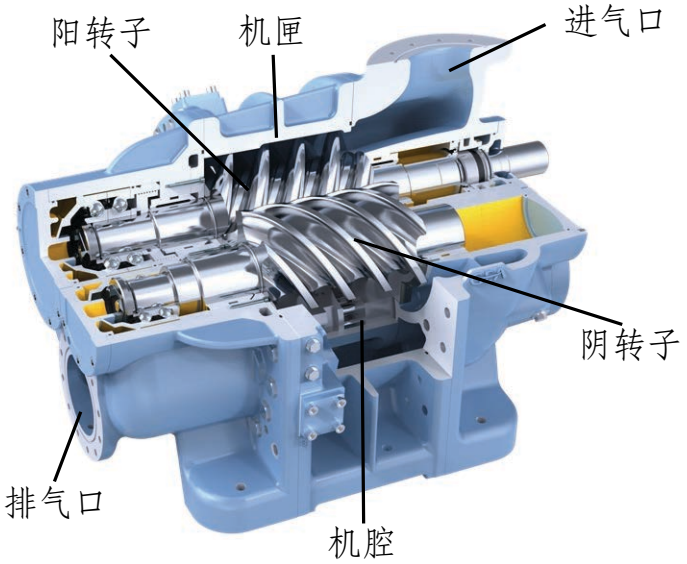


图 6.1 双螺杆旋转压缩机的基本结构[1]
Fig. 6.1 Basic Structure of a Twin-Screw Rotary Compressor

数值仿真在双螺杆旋转压缩机的性能分析和结构设计优化方面具有至关重要的地位[149,150]. 为进行准确的模拟仿真分析, 建立流体通道精准且动态变化的几何模型是其中至关重要的一步. 它不仅对后续的仿真分析具有重要意义, 也是进行结构设计优化的必要前提. 该几何模型涉及到两个相对旋转的转子以及机匣之间的空间关系.

生成高质量结构化网格是对双螺杆压缩机模拟分析的一个关键预处理步骤, 但这一过程面临着极大的挑战. 主要有两方面的因素: 首先, 双螺杆压缩机本身具有复杂的几何形状, 包括极端的纵横比、狭窄的转子间隙、以及急剧变化的间隙尺寸, 这些都极大地增加了构建高质量网格的难度; 其次, 为了准确地描述这些复杂的几何形状, 需要

[1]图源: https://www.gascompressors.co.uk/technologies/oil-flooded-screw-compressor/.

使用大量的网格节点, 特别是在微小间隙以及转子和机匣之间的空间, 这无疑增加了计算的复杂度.

为了应对这些问题, 目前最佳的解决方案是采用专门为双螺杆旋转压缩机设计的网格生成工具, 比如 TwinMesh ™[151] 和 SCORG ™[152]. 这些软件工具能够针对流体和固体计算域的预定时间节点生成一系列计算网格, 从而便于后续使用商业仿真软件 (如 ANSYS CFX ™[153]) 进行仿真分析. 值得一提的是, 这些专业网格生成工具通常只能够生成由直边单元组成的线性网格.

在本章中, 我们特别关注如何从转子轮廓和机匣的点云数据出发, 准确地为双螺杆压缩机的流体部分生成基于样条的体参数化表示. 该方法的核心部分是将第 5 章中所提出的基于椭圆 PDE 参数化方法的进一步拓展和在实际工业应用场景中的验证. 此外, 通过不同的离散方式, 可以很容易的生成具有边界层性质和流对齐的结构化六面体网格. 这种高质量网格不仅便于后续的仿真分析, 而且有助于提高旋转式双螺杆压缩机在模拟和仿真分析中的精度和计算效率.

6.2 主要工作流程

本章中, 我们的出发点是两个转子和机匣的点云数据, 即阳转子 $\{\mathbf{r}_i^{\text{male}}\}$、阴转子 $\{\mathbf{r}_i^{\text{female}}\}$、和机匣的 $\{\mathbf{c}_i\}$. 目标是开发一种基于样条表示的适合分析高质量体参数化:

$$\begin{aligned} &\boldsymbol{x} : \hat{\Omega} \to \Omega \\ &(\xi, \eta, \zeta) \mapsto (x, y, z) = \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \sum_{i_3=0}^{n_3} \mathbf{P}_{i_1,i_2,i_3} N_{i_1,i_2,i_3}(\xi, \eta, \zeta), \end{aligned} \tag{6.1}$$

其中, $\mathbf{P}_{i_1,i_2,i_3}$ 表示控制顶点, $N_{i_1,i_2,i_3}(\xi, \eta, \zeta)$ 是三变量 B 样条基函数, $\hat{\Omega} = [0, 1]^3$ 是参数域, Ω 是物理域.

双螺杆压缩机流体通道的复杂几何特性, 如狭窄的间隙和快速变化的缝隙尺寸, 增加了构建高质量体参数化的难度. 因此, 我们采用一种 2.5D 参数化策略: 在初始阶段, 我们针对每一个平面切片 $\zeta = \zeta_{i_3}$ 单独构建一个适合分析的参数化模型 $\boldsymbol{x}(\xi, \eta, \zeta_{i_3})$. 随后, ζ 参数方向将这些平面切片进行放样操作, 以完成整个参数化过程. 为了表述上的清晰, 图 6.2 展示了本章方法的详细步骤, 其具体流程如下:

1) **预处理步骤**: 将阳转子与阴转子分别视作独立的转子, 并分别与其对应的机匣圆进行参数匹配. 具体来说, 我们将初始机匣点云数据 $\{\mathbf{c}_i\}$ 一分为二, 并将其填补成完整的圆环. 考虑到转子与相应机匣圆之间参数匹配质量对整体参数化结果的显著影响, 采用了外边界正则化子程序以确保参数化结果的质量;
2) **生成基 “O” 型参数化**: 在成功构建出两个独立转子到各自机匣的边界表示后, 利用第 5 章中的方法生成适合分析的高质量 “O” 形参数化;

3) **为每个平面切片的间隙生成高质量参数化**: 对每一平面切片 $\zeta = \zeta_{i_3}$, 在 CUSP 点处对 B 样条曲面进行切割, 进而获得两个独立的 “C” 形参数化. 而对于间隙处, 重新执行参数匹配, 随后重新执行椭圆参数化;

4) **利用放样技术生成最终的体参数化模型**: 将所有切面沿着 ζ 参数方向黏合在一起, 最终形成一个体参数化模型.

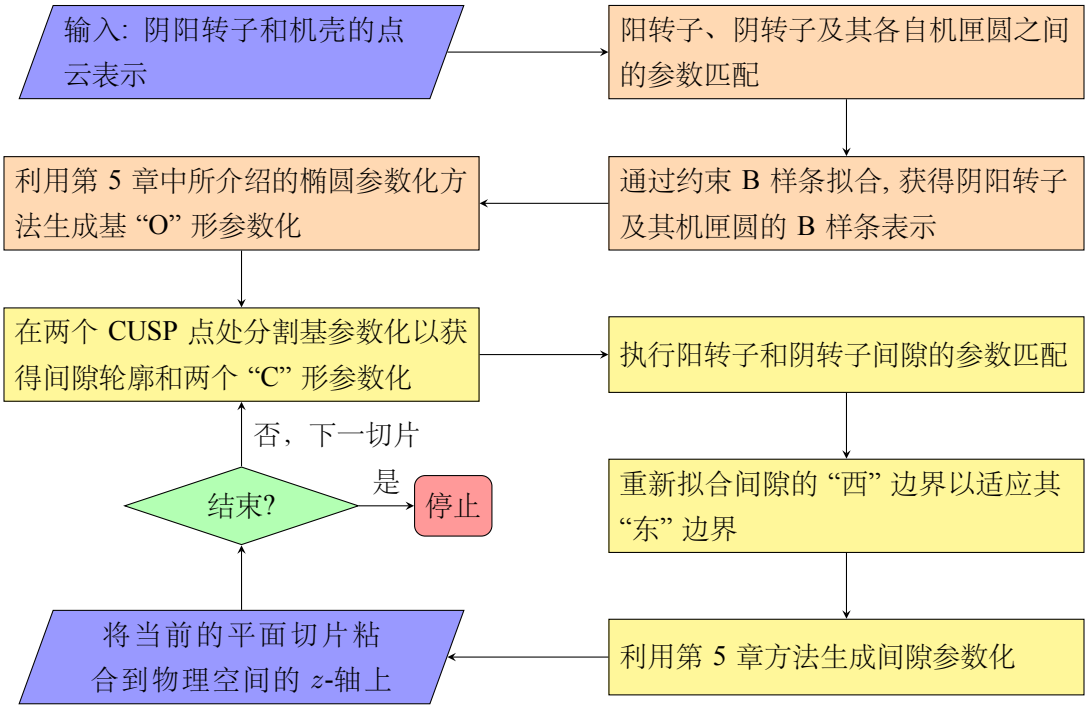


图 6.2　本章方法的主要工作流程

Fig. 6.2　Main Workflow of the Methodology Presented in This Chapter

6.3　参数化方法

6.3.1　预处理步骤

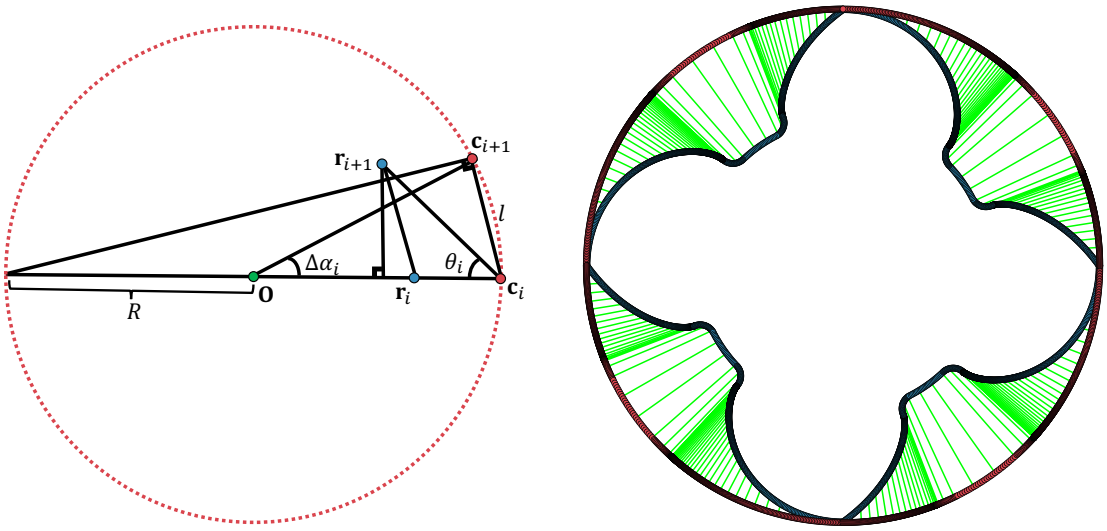
为了实现高质量的参数化, 在将点云转换为 B 样条表示之前, 必须将阳转子和阴转子的参数与其各自的机匣重新匹配. 在这一过程中, 我们把阳转子和阴转子分别当作两个独立的实体进行处理. 原始机匣点云在 CUSP 点处被划分为两个部分, 然后填补成两个完整的圆作为转子各自的机匣圆. 图 6.3(b) 展示了阳转子和其机匣圆.

我们以阳转子为例来说明这一过程, 阴转子也采用完全一致的思路处理. 图 6.3(a) 展示了这一方法的示意图. 首先, 我们给转子轮廓上极角为 0 的点 $\mathbf{r}_0$ 赋一个初始参数值 0. 现在假设已知点 $\mathbf{r}_i$ 及其机匣上对应点的参数值 α_i, 我们来计算转子轮廓上下一个点 $\mathbf{r}_{i+1}$ 的参数值 α_{i+1}. 这里的基本思路是将线段 $\overline{\mathbf{r}_i\mathbf{r}_{i+1}}$ 投影到其对应的机匣圆上. 利用余

弦定理, 我们有以下关系

$$\cos\theta_i = \frac{|\overline{\mathbf{r}_i\mathbf{c}_i}|^2 + |\overline{\mathbf{r}_{i+1}\mathbf{c}_i}|^2 - |\overline{\mathbf{r}_i\mathbf{r}_{i+1}}|^2}{2\,|\overline{\mathbf{r}_i\mathbf{c}_i}|\,|\overline{\mathbf{r}_{i+1}\mathbf{c}_i}|}, \tag{6.2}$$

其中 $|\overline{\mathbf{r}_i\mathbf{c}_i}|$ 表示线段 $\overline{\mathbf{r}_i\mathbf{c}_i}$ 的长度, $|\overline{\mathbf{r}_{i+1}\mathbf{c}_i}|$ 表示线段 $\overline{\mathbf{r}_{i+1}\mathbf{c}_i}$ 的长度, $|\overline{\mathbf{r}_i\mathbf{r}_{i+1}}|$ 表示线段 $\overline{\mathbf{r}_i\mathbf{r}_{i+1}}$ 的长度.



(a) 外边界正则化方法　　(b) 参数匹配结果

图 6.3　外边界正则化方法和阳转子的参数匹配结果

Fig. 6.3　Illustration for Outer Boundary Regularization Subroutine and the Result of the Parameter Matching for the Male Rotor

利用 Pythagorean 恒等式, 我们有

$$\sin\theta_i = \sqrt{1-\cos^2\theta_i}. \tag{6.3}$$

那么投影 $\overline{\mathbf{c}_i\mathbf{c}_{i+1}}$ 的长度为

$$l = \frac{R\cdot|\overline{\mathbf{r}_{i+1}\mathbf{c}_i}|\cdot\sin\theta_i}{R-|\overline{\mathbf{r}_i\mathbf{c}_i}|}. \tag{6.4}$$

由圆心角等于同一弧所对的圆周角的二倍有

$$\Delta\alpha_i = 2\ \arcsin\left(\frac{l}{2R}\right). \tag{6.5}$$

因此, 阳转子上点 $\mathbf{r}_{i+1}$ 所对应的的参数值为

$$\alpha_{i+1} = \alpha_i + \Delta\alpha_i, \tag{6.6}$$

并且其对应机匣圆上的点为

$$\mathbf{c}_{i+1} = (R\cos(\alpha_{i+1}), R\sin(\alpha_{i+1})). \tag{6.7}$$

利用转子几何的对称性可以进一步减少计算量, 这一外边界正则化子程序只在 4 瓣阳转子极坐标下区间 $[0, \pi/2]$ 段上执行. 最后, 将得到的参数值进行归一化处理将其映射到区间 $[0, 1]$ 内, 即 $t_i = \alpha_i / \max_i \alpha_i$. 图 6.3(b) 展示了执行参数匹配后的结果.

在得到参数值 $\{t_i\}_{i=0}^m$ 和转子轮廓上的点 $\{\mathbf{r}_i\}_{i=0}^m$ 以及机匣圆上的点 $\{\mathbf{c}_i\}_{i=0}^m$ 后, 接着采用约束 B 样条拟合将点云数据转换为 B 样条形式. 此时, 为避免非闭合的曲线表示并保留点云的形状特征, 与参数 $t_0 = 0$ 和 $t_m = 1$ 相对应的端点尤为重要. 为实现这一目标, 我们采用具有端点约束的 B 样条最小二乘拟合, 即 $\mathcal{C}(t_0) = \mathbf{r}_0$ 和 $\mathcal{C}(t_m) = \mathbf{r}_m$. 对于一组控制点 $\{\mathbf{P}_i\}_{i=0}^n$ 和点 $\{\mathbf{r}_i\}_{i=0}^m$, 需要求解以下带约束的二次规划问题:

$$\begin{aligned} \arg\min_{\mathbf{P}} \ \hat{E}(\mathbf{r}_i) &= \frac{1}{2}\sum_{j=1}^{m-1} \|\mathcal{C}(t_j) - \mathbf{r}_j\|_2^2, \\ \text{s.t.} \quad \mathcal{C}(t_0) &= \mathbf{r}_0, \\ \mathcal{C}(t_m) &= \mathbf{r}_m, \end{aligned} \tag{6.8}$$

其中 $\mathcal{C}(t_j) = \sum_{i=0}^n \mathbf{P}_i N_i(t_j)$ 表示参数 t_j 处 B 样条曲线上的点, $N_i(\xi)$ 是单变量 B 样条基函数.

考虑问题 (6.8) 的 Karush-Kuhn-Tucker 条件, 控制点 $\{\mathbf{P}_i\}_{i=0}^n$ 的解可以通过求解以下线性系统得到

$$\left[\begin{array}{c|c} \mathbf{A}^{\mathrm{T}}\mathbf{A} & \mathbf{I}_{2\times 2} \\ \hline \mathbf{I}_{2\times 2} & \mathbf{O} \end{array}\right] \left[\begin{array}{c} \mathbf{P}_f \\ \hline \mathbf{P}_c \end{array}\right] = \left[\begin{array}{c} \mathbf{A}^{\mathrm{T}}\mathbf{r}_f \\ \hline \mathbf{r}_c \end{array}\right], \tag{6.9}$$

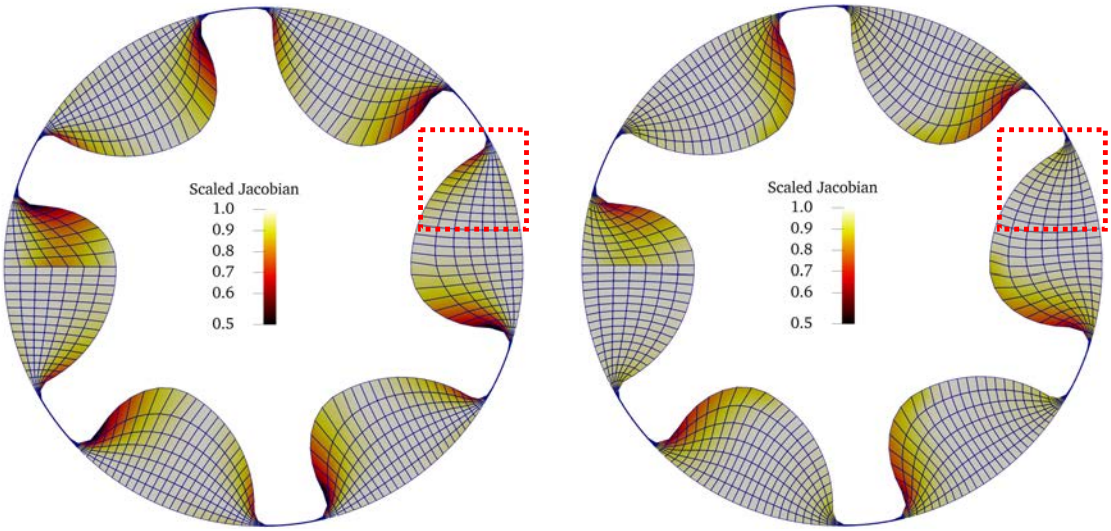
其中, $\mathbf{P}_f = [\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_{n-1}]^{\mathrm{T}}$ 和 $\mathbf{P}_c = [\mathbf{P}_0, \mathbf{P}_n]^{\mathrm{T}}$ 表示自由和约束控制点, $\mathbf{r}_f = [\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_{m-1}]^{\mathrm{T}}$ 和 $\mathbf{r}_c = [\mathbf{r}_0, \mathbf{r}_m]^{\mathrm{T}}$ 表示转子或外壳圆上相应的自由和约束点, $\mathbf{I}_{2\times 2}$ 是单位阵, 且

$$\mathbf{A} = \begin{bmatrix} N_1(t_1) & N_2(t_1) & \cdots & N_n(t_1) \\ N_1(t_2) & N_2(t_2) & \cdots & N_n(t_2) \\ \cdots & \cdots & \ddots & \cdots \\ N_1(t_{m-1}) & N_2(t_{m-1}) & \cdots & N_n(t_{m-1}) \end{bmatrix}$$

是具有 $m-1$ 行和 $n-1$ 列的配置矩阵. 线性系统 (6.9) 中的系数矩阵是稀疏且对称的, 因此可以使用 Cholesky 分解求解器高效地求解该线性系统.

6.3.2 基 “O” 形参数化生成

在得到边界 B 样条曲线表示后, 下一步是为两个转子分别生成适合分析的基 “O” 形参数化. 采用第 5 章中的方法构造的参数化结果如图 6.4 所示. 可以看到, 缩放的 H^1 离散方法获得的参数化质量相较于原先采用的 H^2 离散方法[72] 有了显著的提升. 尤其值得关注的是, 新的参数化方法不仅整体上更加均匀, 而且在接近边界的区域, 其正交性质也有了明显的优化提升.



(a) H^2 离散参数化结果　　(b) 缩放 H^1 离散参数化结果

图 6.4　阴转子: 两种椭圆参数化方法得到的结果

Fig. 6.4　Female Rotor: Parameterization Generated by two Elliptic Parameterization Methods

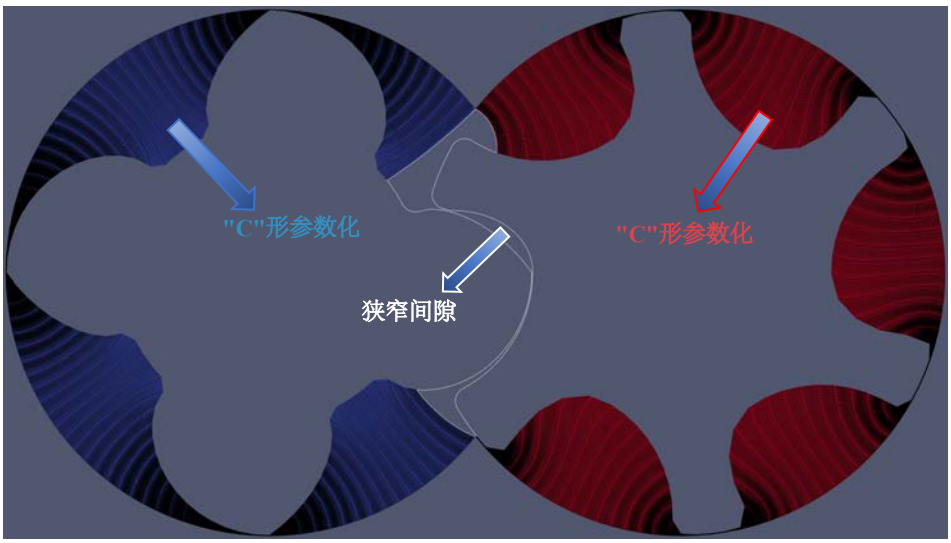


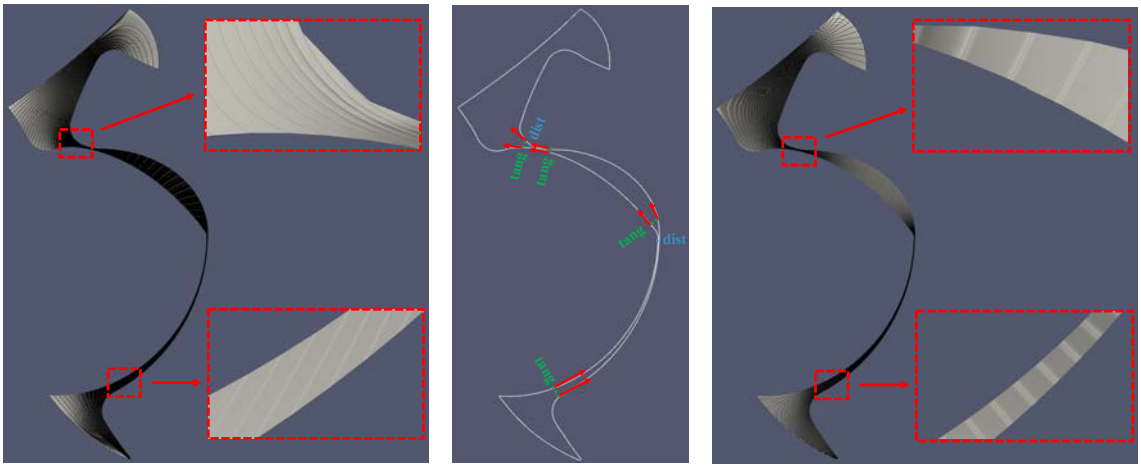
图 6.5　分割基 “O” 形参数化以获得外部表示和狭窄间隙

Fig. 6.5　C-type Parameterizations and Clearance Profile by Splitting Base O-type Parameterization

接下来, 我们将两个转子的基 “O” 形参数化放在一起, 并在两个 CUSP 点处利用 B 样条节点插入算法将其分割开, 以得到两个 “C” 形参数化的外部区域和狭窄间隙的轮廓表示, 如图 6.5 所示. 下面我们将重点重点关注每一片上狭窄间隙的参数化问题.

6.4 狭窄间隙的参数化构造

由于 B 样条曲面的张量积结构, 西侧和东侧边界之间参数的不当匹配会导致参数化质量不佳甚至产生自相交, 如图 6.6(a) 所示. 为解决这一问题, 我们需要重新匹配两边界曲线上的点, 尤其是在狭小间隙附近. 本小节将探讨边界曲线的参数匹配问题.



(a) 参数匹配前的参数化结果　　(b) 边界参数匹配方法　　(c) 参数匹配后的参数化结果

图 6.6　边界参数匹配方法和参数匹配前后的参数化结果

Fig. 6.6　Boundary Parameter Matching and Parameterization Results Before and After Matching

这里的基本想法是固定当前切片的东侧边界, 并允许沿着西侧边界改变参数速度. 首先, 对当前切片的东侧边界进行均匀采样, 并确定西侧边界上的最近点. 此时靠近小间隙的点的参数值非常关键, 我们将距离小于给定公差 $\varepsilon_{\text{dist}}$(实验中设置为 1.0) 的点对选为候选点对. 然后我们找到候选点对间距离的局部最小值, 即满足条件 $\text{dist}_{i-1} \leq \text{dist}_i \leq \text{dist}_{i+1}$ 的点对. 进一步地, 为提高参数匹配效果, 我们额外考虑候选点对的切方向. 具体来说, 首先计算每个候选点对上的规一化切线向量, 如果它们满足 $\text{tang}_i < \varepsilon_{\text{tang}} < \text{tang}_{i+1}$ 或 $\text{tang}_i > \varepsilon_{\text{tang}} > \text{tang}_{i+1}$, 则将其作为匹配点对. 考虑到算法的稳定性, 即满足 Schoenberg-Whitney 条件, 我们不允许相邻的节点值过近. 图 6.6(b) 展示了参数匹配过程. 利用约束最小二乘方法得到狭小间隙的边界曲线表示后, 再次使用椭圆参数化方法对间隙进行参数化, 参数化结果展示在图 6.6(c) 中. 显然, 与未经重新参数化的参数化结果相比, 重新参数化边界曲线极大地提高了参数化质量.

利用上述讨论得到的转子旋转时平面参数化的演变过程如图 6.7 所示.

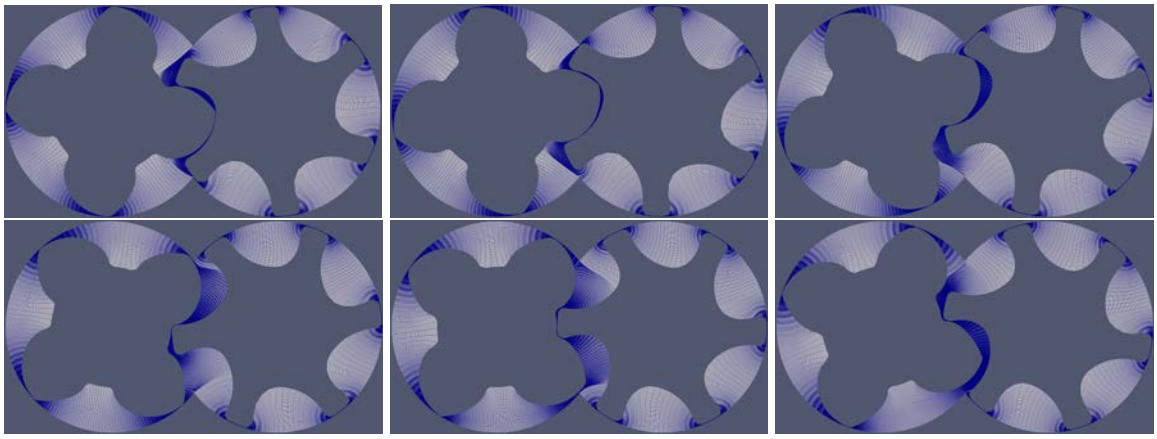
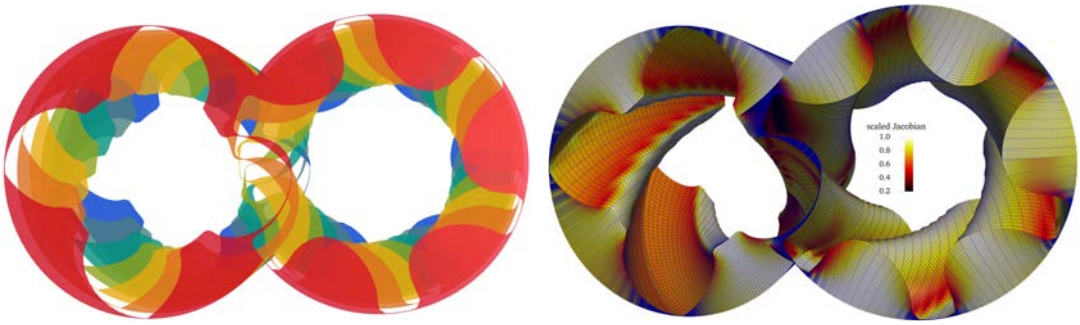


图 6.7　转子旋转时平面参数化的演变
Fig. 6.7　Evolution of the planar parameterizations upon rotation of the rotors

接着将得到的 B 样条平面参数化沿着 ζ-参数方向粘合, 如图 6.8(a) 所示. 最后, 利用 B 样条放样技术得到的体参数化结果如图 6.8(b) 所示. 这里, 沿着 ζ-参数方向分为 61 个平面切片. 值得一提的是, 整个计算流程在 MacBook Pro 14-inch 2021, 配备 Apple M1 Pro CPU 和 16GB RAM 的环境下 3 秒内完成. 这突显了所提出方法的高效性.



(a) 沿 ζ-参数方向的一族平面参数化　　(b) 体参数化结果

图 6.8　沿 ζ-参数方向的一族平面参数化和体参数化结果
Fig. 6.8　Planar Parameterizations along ζ-Direction and Volumetric Parameterization

6.5　网格离散化技术

样条体参数化的另一个优势在于: 通过采用灵活的离散化策略, 我们不仅可以便捷地生成具有不同性质的网格, 还能定制网格以满足特定的分析需求. 在本节中, 我们将特别关注如何利用这一特性来创建边界层网格和与流场对齐的网格. 这些示例将进一步说明样条体参数化方法在复杂分析场景中的多样性和灵活应用能力.

这里的基本思路是引入一个参数变换 $\boldsymbol{\xi}(\hat{\boldsymbol{\xi}}) : [0,1]^3 \to [0,1]^3$. 值得说明的是, 如果原来的样条体参数化 $\boldsymbol{x}$ 和新引入的参数变换 $\boldsymbol{\xi}(\hat{\boldsymbol{\xi}})$ 都是是单值的, 那么经过充分稠密采样得到的结构六面体网格 $\{\boldsymbol{x} \circ \boldsymbol{\xi}(\hat{\boldsymbol{\xi}}_{i,j,k})\}$ 一定是单值的.

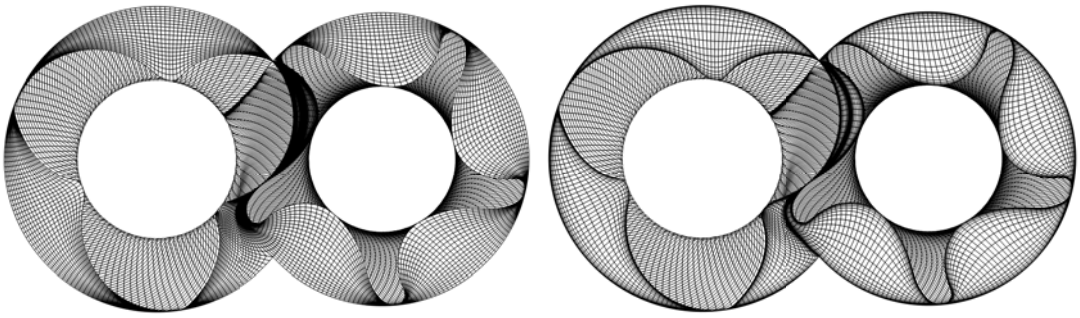
6.5.1 边界层网格

在流体力学计算仿真中, 解 (例如流速或压力) 通常会在边界附近急剧变化. 使用边界层网格有助于精确捕捉这些在边界附近的突变现象. 这种网格不仅在与物体表面平行的方向上有更好的几何适应性, 而且还可以通过调整网格层数量, 在垂直于物体表面的方向上精细地捕捉流场的细微变化.

幸运的是, 一旦我们获取了一个高质量的样条体参数化, 就能轻易地将其转换为带有边界层特性的结构六面体网格. 在这里, 我们只需要进行一次重新采样即可. 具体来说, 引入如下参数变换

$$\begin{cases} \xi = \hat{\xi}, \\ \eta = \dfrac{\tanh\left(\alpha(2\hat{\eta}-1)\right)}{2\tanh(\hat{\eta})} + \dfrac{1}{2}, \\ \zeta = \hat{\zeta}, \end{cases} \tag{6.10}$$

其中 α 表示伸缩因子, 用来调整边界层网格的密度. 利用恒同映射生成的结构六面体网格如图 6.9(a) 所示, 而当我们选择 $\alpha = 1$ 时, 由变换 (6.10) 生成的网格将具有明显的边界层性质, 如图 6.9(b) 所示.



(a) 均匀六面体网格　　(b) 边界层六面体网格

图 6.9　利用参数变换生成具有边界层特性的结构六面体网格

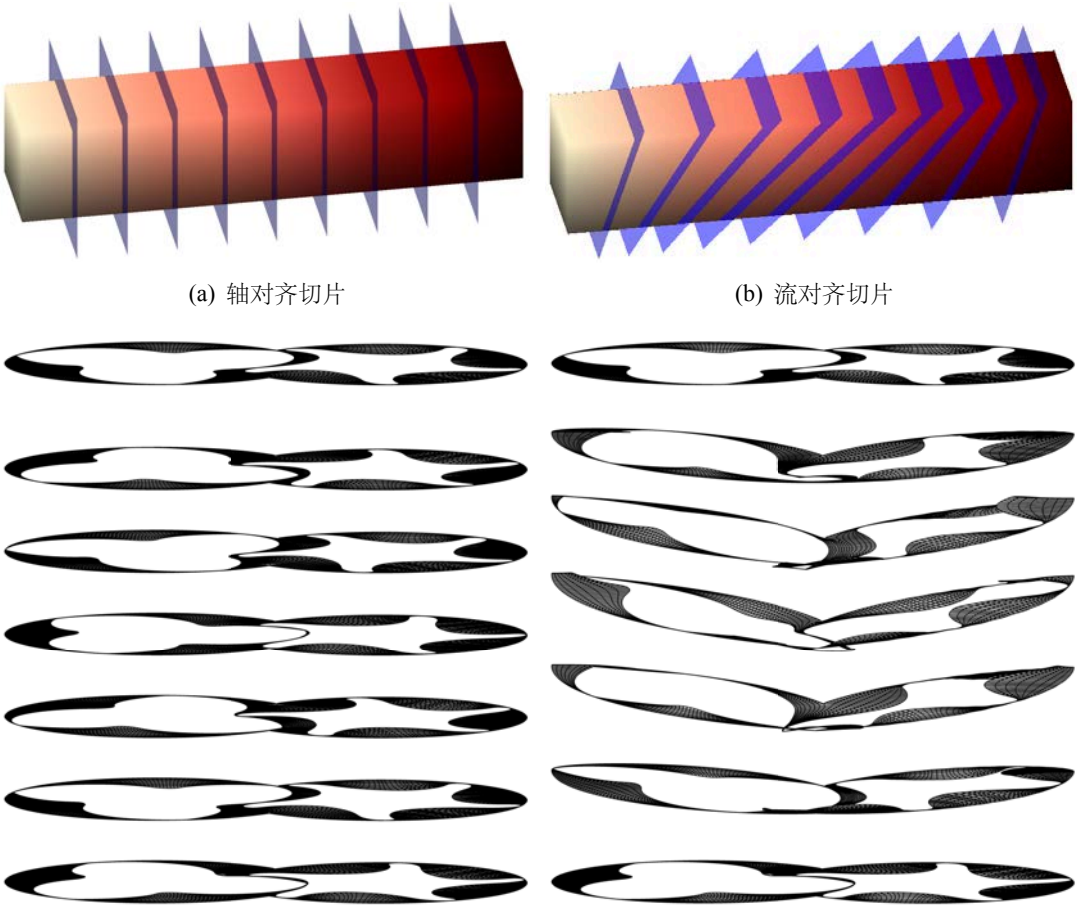
Fig. 6.9　Generating Structured Hexahedral Meshes with Boundary Layer via Parameter Transformation

6.5.2 流对齐网格

Kovacevic 和 Rane 指出如果网格与流场方向完全对齐, 可以有效地缓解数值扩散[111]. 反之, 即使网格质量优良或离散化精度高, 与流场方向的不对齐也将不可避免地导致数值扩散. 值得注意的是, 这种现象不仅在结构网格中普遍存在, 非结构网格也同样面临这一问题.

为解决这一问题, 最佳的选择是使用与齿条正交的结构贴体网格. 这种网格设计会在沿着转子螺旋的方向上具有二维横截面, 以更准确地捕捉流场特性. 而从体样条

参数化出发, 我们仅需要引入一个参数变换即可得到与流方向对齐的结构六面体网格. 而且这一方法的单值性具有理论保证.



(a) 轴对齐切片 (b) 流对齐切片

(c) 轴对齐网格 (d) 流对齐网格

图 6.10 利用参数变换生成流对齐的网格

Fig. 6.10 Generating Flow-aligned Meshes with Boundary Layer via Parameter Transformation

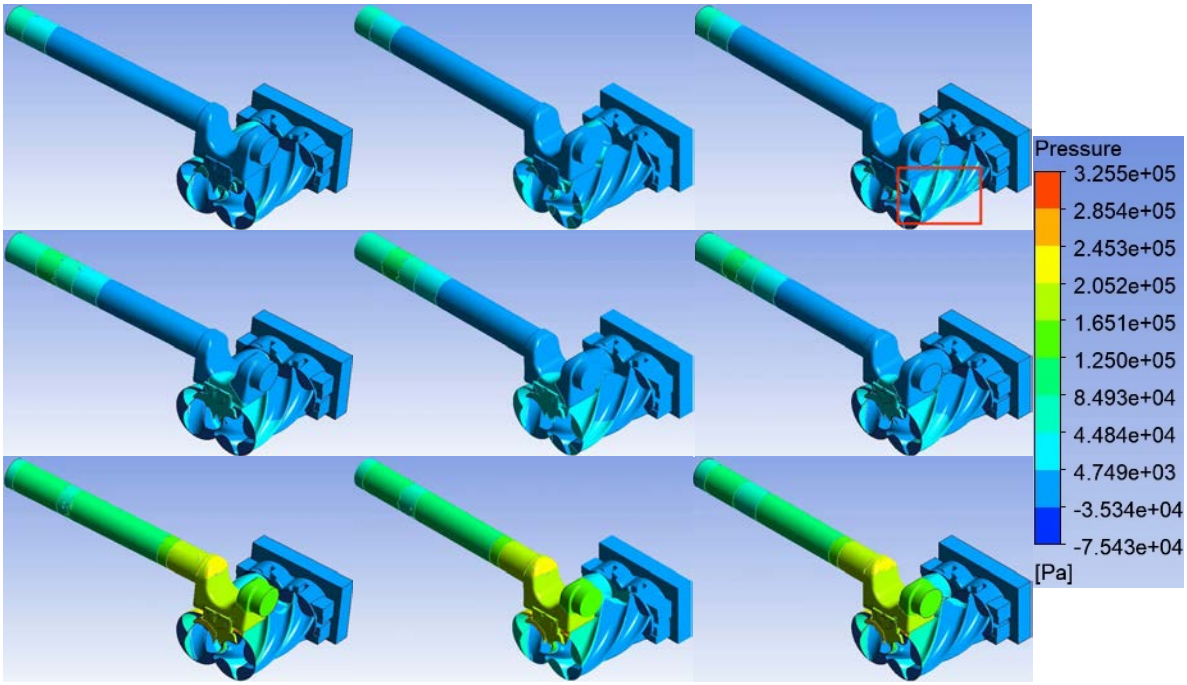
具体地, 我们在 ζ-参数方向引入如下参数变换 $\zeta = \hat{\zeta} + \beta\ \sin(\pi\hat{\xi})\,\overline{h}$: 其中,

$$\overline{h} = \begin{cases} -\xi & \text{如果}\xi \le \frac{1}{2}, \\ \xi - 1 & \text{如果}\xi > \frac{1}{2} \text{ and } \xi \le \frac{3}{4}, \\ 1 - \xi & \text{如果}\xi > \frac{3}{4}. \end{cases} \tag{6.11}$$

直观上来讲, 我们沿着 ζ 参数方向采用如图 6.10(d) 所示的一种特殊的切片策略, 得到的流对齐结构六面体网格如图 6.10(b) 所示. 相比之下, 轴向对齐的切片策略和相应的六面体结构网格分别在图 6.10(a) 和图 6.10(c) 中给出.

6.5.3 流体仿真分析

我们将三种结构六面体网格导入 ANSYS CFX ™ 进行计算流体力学仿真分析, 得到的压力分布图如图 6.11 所示. 这说明本方法得到的结构六面体网格是适合分析的. 值得注意的是, 分析结果的第一个时间步 (参见图 6.11 第一行), 流对齐网格的分析结果与另外两种网格有显著不同, 而在其他时间步上的分析结果仅有微小差异. 与试验数据的详细对比, 值得进一步深入研究. 此外, 为双螺杆压缩机械开发基于等几何分析的数值分析和结构设计优化工具也是一个值得深入研究的方向.



(a) 轴对齐网格　　(b) 边界层网格　　(c) 流对齐网格

图 6.11　不同类型网格在 ANSYS CFX ™ 中的仿真结果

Fig. 6.11　Simulation Results of Different Types of Grids in ANSYS CFX ™

6.6 本章小结

在本章中, 我们成功地将第 5 章中所提出的参数化方法应用到双螺杆压缩机的网格生成问题中. 实验结果表明, 该方法能够高效地生成具有更高质量的体参数化结果. 边界曲线之间的参数匹配对于提高参数化质量有着至关重要的影响. 为此, 我们设计了一种新的参数匹配方法, 该方法显著提升了所生成参数化的质量. 利用不同的网格离散策略, 我们可以方便地生成具有边界层特性的网格, 也能创建与流场方向一致的结构六面体网格. 这些特性使我们的方法具有很高的灵活性和应用价值. 在商用软件 ANSYS CFX ™ 中的测试验证进一步印证了本方法的有效性. 实验结果表明, 本章方法生成的网格能够很好地支持复杂的流体力学仿真分析.

7 基于绝对主曲率度量和双层优化的 r 自适应方法

7.1 引言

在之前的章节中, 我们探讨了各向同性参数化的构造方法. 这些方法要求参数化在满足单值性的前提下, 具有良好的正交性和均匀性. 然而, 这类方法与待求解的偏微分方程无关, 不能针对具体的物理问题进行优化. 对于特定的 PDE, 从数值精度和计算效率的角度来看, 各向同性参数化可能并不是最优选择. 特别是, 在很多物理现象中, 如波的传播、激光打印和应力集中问题等, 局部和各向异性现象是十分普遍的. 针对这类问题, 前述的各向同性参数化在仿真分析方面的计算效率可能会较低. 在过去的十年里, 等几何自适应技术已成为研究的热点问题. 与各向同性参数化不同, 各向异性参数化通常会依赖于待求解的方程, 从而期望能提供更为可靠和高效的计算结果.

许多研究集中在使用具有局部细化能力的样条工具进行 h 自适应, 如层次 B 样条[40,96], 截断层次 B 样条 (THB 样条)[41,97,98], T 样条[42,85,86] 以及截断 T 样条[99]. 这些方法在数值误差较大的区域内增加自由度, 在处理局部问题方面表现出卓越的性能. 另外两种网格自适应方案是基于升阶或降阶的 p 自适应方法以及基于基函数粗化或平滑的 k 自适应方法[11,14], 如具有局部 p 细化能力的混合阶加权 T 样条[100].

另一类可能的解决方案是基于重新参数化的 r 自适应方法. 据作者所知, 在等几何分析背景下, 这类方法可以追溯到徐岗等人提出的 r 细化方法[53]. 与 h 自适应相比, r 自适应方法希望在保持总自由度不变的情况下提高每个自由度的精度.

通过调整内部的控制顶点, 徐岗等人[53] 提出了面向等几何分析的 r 细化方法, 该方法最小化基于后验误差估计的目标函数. 然而, 这种方法无法保证参数化结果的双射性. Gravesen 等人[52] 提出了一个面向分析的参数化方法框架, 但这一方法仅适用于小规模问题. 随后, 徐岗等人[104] 提出了一种将 Winslow 映射与绝对曲率和后验误差估计相结合的 r 自适应方法. 这种方法使用 Winslow 项来保证双射性, 但也因此牺牲了参数化质量. 此外, Ali 和马维银[105] 提出了一种 IGA-Collocation-PDE 方法来构建依赖于解的各向异性参数化方法.

为此, 本章的目的是开发一种高效且鲁棒的 r 自适应参数化方法. 其基本思路是将等几何分析数值解视为 $\mathbb{R}^3$ 中的参数曲面, 并利用绝对主曲率来表征等几何分析解的变化程度. 因此, 该方法建立了几何量和分析结果之间的紧密联系.

本章的主要贡献总结如下:

1) 直接将绝对主曲率度量作为目标函数可能会产生具有自相交的参数化结果. 为此, 我们对目标函数进行修正, 并借助于带有线搜索准则 (如 Wolfe-Powell 条件) 的优化算法, 有效地保证了参数化结果的单值性;

2) 详细推导出了本章所提出的 r 自适应参数化方法的灵敏度传播和解析灵敏度计算公式, 从而增强了基于梯度的优化算法中的数值稳定性;

3) 采用具有相同几何形状而细化程度不同的双层方法, 即我们在粗参数化 (称为设计模型) 层次上更新参数化以提高效率, 以及细参数化 (称为分析模型) 层次上来执行等几何仿真分析以提高计算精度. 基本工作流程如图 7.1 所示. 此外, 我们展示了如何从设计模型到分析模型的灵敏度传播以及解析灵敏度计算的详细算法;

4) 为了验证所提出方法的有效性, 我们在一系列数值实验中对其进行了测试. 结果表明, 与各向同性参数化相比, 本章方法得到的参数化结果在许多情况下能够显著提高计算效率和数值精度. 此外, 我们还发现所提出的方法在处理具有复杂几何和物理特性的问题时表现出了很好的鲁棒性. 最后, 将所提出的方法应用于具有移动 Gaussian 热源的二维线性热传导问题, 这是增材制造应用中的简化模型.

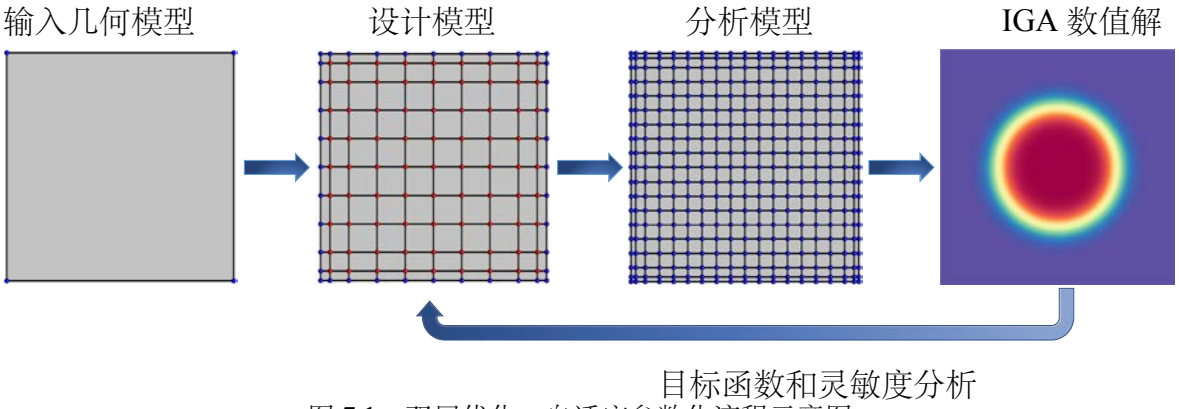


图 7.1　双层优化 r 自适应参数化流程示意图

Fig. 7.1　Schematic Diagram of Bi-Level Optimization for r-Adaptive Parametrization

7.2　基于绝对主曲率度量的 r 自适应方法

本节中, 我们将要介绍基于绝对主曲率度量的 r 自适应参数化方法. 其核心思想是将等几何分析的数值解视为一个三维参数曲面, 然后利用这一参数曲面的曲率信息来刻画解的变化情况.

7.2.1　绝对主曲率

首先, 我们考虑具有如下形式的等几何分析数值解曲面 $\boldsymbol{\mathcal{S}}(\xi,\eta)$:

$$\boldsymbol{\mathcal{S}}(\xi,\eta) = (x(\xi,\eta),\ y(\xi,\eta),\ \mathscr{U}(\xi,\eta)) = \mathbf{R}^{\mathrm{T}}\widetilde{\mathbf{P}}, \tag{7.1}$$

其中 $\widetilde{\mathbf{P}} = [\mathbf{P}, \mathscr{U}]$ 是曲面 $\boldsymbol{\mathcal{S}}(\xi,\eta)$ 的控制顶点, $\mathbf{R}$ 表示由 NURBS 基函数 R_i 组成的向量.

在经典微分几何中, $\boldsymbol{\mathcal{S}}(\xi,\eta)$ 的第一和第二基本形式定义为

$$I=\begin{bmatrix}E & F\\ F & G\end{bmatrix}=\begin{bmatrix}\langle\boldsymbol{\mathcal{S}}_\xi,\ \boldsymbol{\mathcal{S}}_\xi\rangle & \langle\boldsymbol{\mathcal{S}}_\xi,\ \boldsymbol{\mathcal{S}}_\eta\rangle\\ \langle\boldsymbol{\mathcal{S}}_\xi,\ \boldsymbol{\mathcal{S}}_\eta\rangle & \langle\boldsymbol{\mathcal{S}}_\eta,\ \boldsymbol{\mathcal{S}}_\eta\rangle\end{bmatrix} \tag{7.2}$$

和

$$II=\begin{bmatrix}L & M\\ M & N\end{bmatrix}=\begin{bmatrix}\langle\boldsymbol{\mathcal{S}}_{\xi\xi},\ \mathbf{n}\rangle & \langle\boldsymbol{\mathcal{S}}_{\xi\eta},\ \mathbf{n}\rangle\\ \langle\boldsymbol{\mathcal{S}}_{\xi\eta},\ \mathbf{n}\rangle & \langle\boldsymbol{\mathcal{S}}_{\eta\eta},\ \mathbf{n}\rangle\end{bmatrix}, \tag{7.3}$$

其中 $\langle\cdot,\ \cdot\rangle$ 表示两个向量的内积, 且

$$\mathbf{n}=\frac{\boldsymbol{\mathcal{S}}_\xi\times\boldsymbol{\mathcal{S}}_\eta}{\|\boldsymbol{\mathcal{S}}_\xi\times\boldsymbol{\mathcal{S}}_\eta\|} \tag{7.4}$$

表示曲面 $\boldsymbol{\mathcal{S}}(\xi,\eta)$ 的单位法向量.

于是, 我们有如下的 Gauss 曲率

$$K=\frac{LN-M^2}{EG-F^2} \tag{7.5}$$

和平均曲率

$$H=\frac{LG-2MF+NE}{2(EG-F^2)}. \tag{7.6}$$

那么, 解曲面的最大和最小主曲率分别由以下公式计算

$$\kappa_{max}=H+\sqrt{H^2-K} \tag{7.7}$$

和

$$\kappa_{min}=H-\sqrt{H^2-K}. \tag{7.8}$$

最后, 我们可以将绝对主曲率定义为

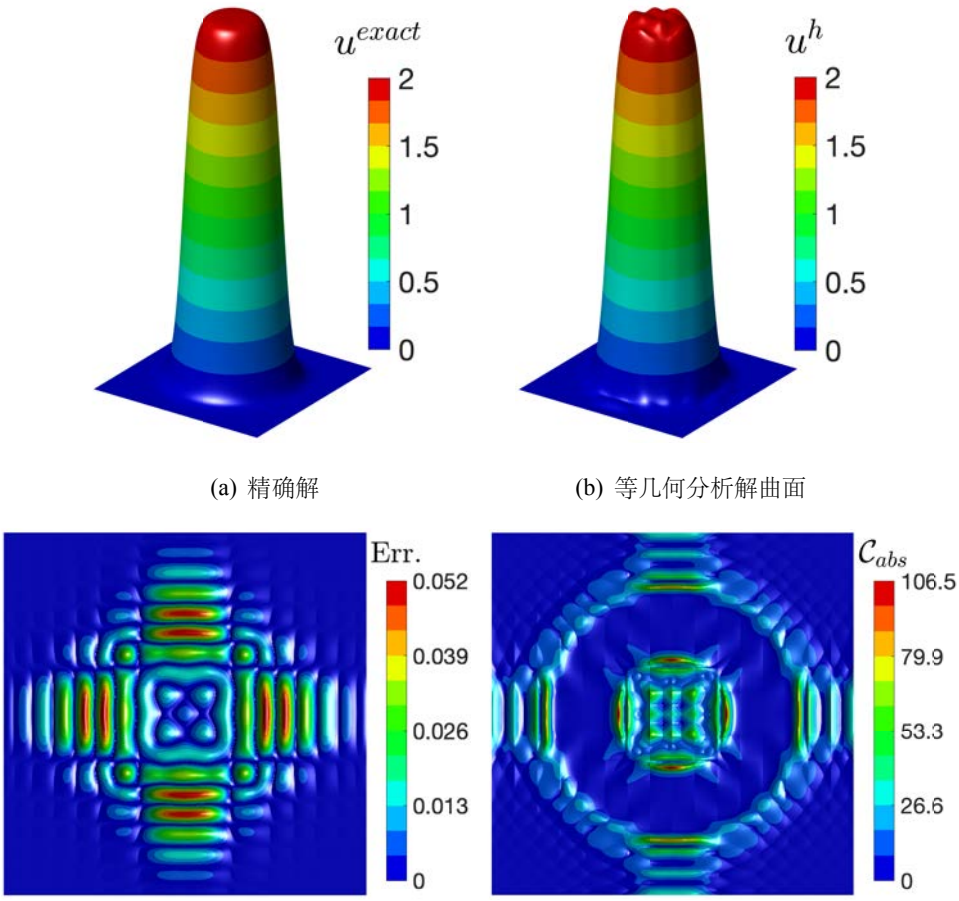
$$\mathcal{C}_{abs}=|\kappa_{max}|+|\kappa_{min}|. \tag{7.9}$$

或者是等价地, 我们有

$$\mathcal{C}_{abs}=\begin{cases}2H & \text{若 } \kappa_{min}\geq 0,\\ 2\sqrt{H^2-K} & \text{若 } \kappa_{max}\geq 0,\\ -2H & \text{其它}.\end{cases} \tag{7.10}$$

在接下来的讨论当中, 为了便于灵敏度分析的计算, 我们更倾向于使用式 (7.10) 而

不是 (7.9). 这是因为使用式 (7.10) 可以有效地降低计算成本. 我们将在7.3节中进一步讨论这些问题.



(a) 精确解　　(b) 等几何分析解曲面

(c) 绝对误差分布　　(d) 绝对主曲率分布

图 7.2　等几何分析解曲面, 数值误差及绝对主曲率分布

Fig. 7.2　IGA Solution Surface, Numerical Error and Absolute Principal Curvature

图 7.2 展示了等几何分析解曲面以及其误差和绝对主曲率分布. 观察可知, 在解变化较大的区域内, 数值误差和解曲面的绝对主曲率均相对较大. 因此, 我们的基本策略是利用式 (7.2) 中的绝对主曲率来识别数值误差较大的区域.

7.2.2　目标泛函

利用逐点绝对主曲率 (7.10), 我们定义如下的绝对主曲率度量

$$
\begin{aligned}
\hat{\Phi}(\boldsymbol{x}) &= \Phi(\boldsymbol{x}, \mathscr{U}(\boldsymbol{x})) \\
&= \begin{cases} \int_{\hat{\Omega}} \mathcal{C}_{abs}\, d\hat{\Omega} & \text{若 } |\boldsymbol{\mathcal{J}}| > 0,\ \forall\, (\xi, \eta) \in \hat{\Omega}, \\ +\infty & \text{其它}. \end{cases}
\end{aligned} \tag{7.11}
$$

参数化结果是适合分析的第一个要求就是满足单值性. 一般的一阶数值优化算法都配备线搜索子程序以确保目标函数有充分的下降. 借助这一特点, 只需要在其违反双射条件时, 将目标函数值 $\Phi(\boldsymbol{x})$ 设为正无穷. 换言之, 当参数化的雅可比行列式在某个 Gauss 积分点处小于或等于 0 时, 即 $|\boldsymbol{\mathcal{J}}| \leq 0$, 那么直接将目标函数值 $\Phi(\boldsymbol{x})$ 取为正无穷. 直观来看, 如果迭代过程中违反了双射性条件, 那么优化求解器将拒绝当前步长, 并重新计算一个满足双射要求的新步长. 本章采用带有强 Wolfe 条件回溯线搜索方法[128] 的标准 L-BFGS 求解器[124].

将内部控制顶点的位置作为优化变量, 基于主曲率度量的 r 自适应参数化方法可以归结为求解以下非线性优化问题

$$\underset{\mathbf{P}_i}{\arg\min} \underset{i\in\mathcal{I}_I}{} \hat{\Phi}(\boldsymbol{x}), \tag{7.12a}$$

$$\text{s.t. } \mathbf{K}(\boldsymbol{x})\, \mathscr{U}(\boldsymbol{x}) = \mathbf{f}(\boldsymbol{x}), \tag{7.12b}$$

其中 $\mathbf{P}_i = (x_i, y_i)\ i \in \mathcal{I}_I$ 是内部的控制顶点.

注 7.1. 在文献[104]中, 徐岗等人采用了如下带有 Winslow 修正项的目标泛函

$$\Phi^{\text{Winslow}} = \int_{\hat{\Omega}} \mathcal{C}_{abs} \frac{x_\xi^2 + x_\eta^2 + y_\xi^2 + y_\eta^2}{|\boldsymbol{\mathcal{J}}|}\, d\hat{\Omega} \tag{7.13}$$

作为目标函数. 当雅可比行列式 $|\boldsymbol{\mathcal{J}}|$ 接近于零时, 积分的后半部分趋向于正无穷, 从而防止结果参数化产生自相交. 我们称这部分为 Winslow 项. 然而, Winslow 项可能会影响最优解并牺牲一定的精度. 我们将在第7.4节中进行比较. 最后, 我们强调, 由于优化过程中目标泛函值的变化并非是连续的, 可能会导致直接从正值跳跃到负值的情况, 从而引起无效的参数化结果[127].

7.3 灵敏度分析

在基于梯度的优化算法中, 解析的梯度计算对计算效率和数值稳定性具有至关重要的影响. 因此, 本节致力于优化问题 (7.12) 的灵敏度分析. 同时考虑到计算效率, 我们采用双层优化策略, 其中分析模型是设计模型的加密版本.

7.3.1 灵敏度传播: 从设计模型到分析模型

NURBS 几何细化: 首先我们介绍 NURBS 几何的细化. 等几何分析中有三种细化策略, 即基于 NURBS 节点插入算法的 h 细化、基于 NURBS 升阶算法的 p 细化和将节点插入和升阶相结合的 k 细化方法. 值得注意的是, 这里所有的细化方法所得到的细化模型上的控制顶点都仅是较粗模型控制顶点的线性组合. 具体来说, 细化后模型的

控制顶点 $\mathbf{Q}$ 通过以下方式计算

$$\mathbf{Q} = \mathsf{Refine} \cdot \mathbf{P}, \tag{7.14}$$

其中 Refine 是一个 $m \times n$ 稀疏矩阵, 我们称它为细化算子, m 是细化后模型控制顶点的数量, $\mathbf{P}$ 是粗模型所有控制顶点 (n 个) 所形成的向量.

NURBS 曲面的细化是通过对其齐次坐标形式进行细化来实现的. 具体来说, 有三个步骤: 将给定的 NURBS 曲面转化为高一维空间中的 B 样条曲面, 然后对该 B 样条曲面进行细化, 最后将计算出来的控制顶点投影至原先的空间中. 形式上, 细化后 NURBS 模型的控制顶点 $\mathbf{Q}$ 通过以下线性变换计算得出

$$\mathbf{Q} = \widetilde{\mathsf{Refine}} \cdot \mathbf{P}, \tag{7.15}$$

其中 $\widetilde{\mathsf{Refine}} = diag(\boldsymbol{\omega}_{\mathbf{Q}})^{-1}\,\mathsf{Refine}\,diag(\boldsymbol{\omega}_{\mathbf{P}})$, $diag(\boldsymbol{\omega}_{\mathbf{P}})$ 是对角元素为粗模型上权因子的 $n \times n$ 对角矩阵, $diag(\boldsymbol{\omega}_{\mathbf{Q}})$ 是对角元素为细化后模型上权因子的 $m \times m$ 对角矩阵. 值得注意的是, 如果所有权重 $\boldsymbol{\omega}_{\mathbf{P}}$ 都相等, 则细化关系式 (7.15) 将退化为式 (7.14).

灵敏度传播: 在本章的问题中, 内部控制顶点的位置被用作设计变量 (优化变量). 因此, 对于设计模型的控制顶点 $\mathbf{P}$ 来说, 其关于设计变量的导数为

$$\begin{aligned} \frac{\partial \mathbf{P}}{\partial x_s} &= [\mathbf{0},\ \ldots,\ \mathbf{0},\ \underset{s\text{-th}}{\mathbf{e}_1},\ \mathbf{0},\ \ldots,\ \mathbf{0}]^{\mathrm{T}}, \\ \frac{\partial \mathbf{P}}{\partial y_s} &= [\mathbf{0},\ \ldots,\ \mathbf{0},\ \underset{s\text{-th}}{\mathbf{e}_2},\ \mathbf{0},\ \ldots,\ \mathbf{0}]^{\mathrm{T}}, \end{aligned} \tag{7.16}$$

其中 $\mathbf{e}_1 = [1, 0]^{\mathrm{T}}$, $\mathbf{e}_2 = [0, 1]^{\mathrm{T}}$, 这里符号 $(\ldots)$ 表示省略的元素全部被置为 0. 在本章的余下部分, 为了符号上的一致, 我们使用 α 表示 x 和 y, 而不再区分它们.

对式 (7.15) 进行求导, 我们得到:

$$\frac{\partial \mathbf{Q}}{\partial \alpha_s} = \widetilde{\mathsf{Refine}} \frac{\partial \mathbf{P}}{\partial \alpha_s}. \tag{7.17}$$

这意味着一旦计算出了细化关系式 (7.15) 和设计模型的导数 (7.16), 我们可以直接通过式 (7.17) 得到分析模型控制顶点关于设计变量 α_s 的导数 $\partial\mathbf{Q}/\partial\alpha_s$. 在计算出 $\partial\mathbf{Q}/\partial\alpha_s$ 之后, 灵敏度分析的余下步骤主要在分析模型上进行.

最后, 需要注意的是, 一旦确定了设计和分析模型的细化次数, 细化关系式 (7.15) 和导数 (7.16) 就不再改变. 为了提高计算效率, 我们建议在优化之前预计算 $\partial\mathbf{Q}/\partial\alpha_s$, 并在迭代过程中重复使用. 为了清晰起见, 我们将从设计模型到分析模型的灵敏度传播过程总结在算法7.1中.

Algorithm 7.1: 从设计模型到分析模型的灵敏度传播

Input: $\mathbf{P}$: 设计模型的控制顶点,
r_h, r_p: h 和 p 细化的次数
n_{desg}: 设计模型内部控制顶点的个数;

Output: $\mathbf{Q}$: 分析模型的控制顶点,
$\partial\mathbf{Q}/\partial\alpha_s$: 分析模型控制顶点关于优化变量的灵敏度;

// 执行 r_p 次 p 细化和 r_h 次 h 细化

计算细化矩阵 $\widetilde{\mathsf{Refine}}$;

计算分析模型的控制顶点 $\mathbf{Q} = \widetilde{\mathsf{Refine}} \cdot \mathbf{P}$; // (7.15)

for $s = 1; s \leq n_{desg}; s++$ **do**

  计算 $\partial\mathbf{P}/\partial\alpha_s$; // (7.16)

  计算 $\partial\mathbf{Q}/\partial\alpha_s = \widetilde{\mathsf{Refine}} \cdot \partial\mathbf{P}/\partial\alpha_s$; // (7.17)

end

7.3.2 灵敏度分析

一旦计算出分析模型的控制顶点关于优化变量的导数 $\partial\mathbf{Q}/\partial\alpha_s$, 随后的灵敏度分析将主要在分析模型上进行计算.

为了符号上的清晰, 我们记分析模型上的几何参数化和解曲面分别为

$$\boldsymbol{x} = \mathbf{R}_A^{\mathrm{T}}\mathbf{Q} \tag{7.18}$$

和

$$\boldsymbol{\mathcal{S}} = \mathbf{R}_A^{\mathrm{T}}\widetilde{\mathbf{Q}}, \tag{7.19}$$

其中 $\mathbf{R}_A$ 是分析模型的 NURBS 基函数, $\widetilde{\mathbf{Q}} = [\mathbf{Q}, \mathscr{U}]$ 是等几何分析解曲面的控制顶点.

利用链式法则, 我们有绝对主曲率度量 (7.12a) 的灵敏度

$$\begin{aligned}\frac{\partial\hat{\Phi}(\boldsymbol{x})}{\partial\alpha_s} &= \frac{\partial\Phi(\boldsymbol{x}, \mathscr{U}(\boldsymbol{x}))}{\partial\alpha_s} \\ &= \frac{\partial\Phi(\boldsymbol{x}, \mathscr{U}(\boldsymbol{x}))}{\partial\alpha_s} + \frac{\partial\Phi(\boldsymbol{x}, \mathscr{U}(\boldsymbol{x}))}{\partial\mathscr{U}}\frac{\partial\mathscr{U}(\boldsymbol{x})}{\partial\alpha_s}.\end{aligned} \tag{7.20}$$

对离散平衡方程 (7.12b) 求导, 我们得到

$$\frac{\partial\mathbf{K}(\boldsymbol{x})}{\partial\alpha_s}\mathscr{U}(\boldsymbol{x}) + \mathbf{K}(\boldsymbol{x})\frac{\partial\mathscr{U}(\boldsymbol{x})}{\partial\alpha_s} = \frac{\partial\mathbf{F}(\boldsymbol{x})}{\partial\alpha_s}. \tag{7.21}$$

经过一些简单的计算, 有

$$\frac{\partial \mathscr{U}(\boldsymbol{x})}{\partial \alpha_s} = \mathbf{K}(\boldsymbol{x})^{-1}\left(\frac{\partial \mathbf{F}(\boldsymbol{x})}{\partial \alpha_s} - \frac{\partial \mathbf{K}(\boldsymbol{x})}{\partial \alpha_s}\mathscr{U}(\boldsymbol{x})\right). \tag{7.22}$$

上式 (7.22) 中右端项

$$\mathbf{f}^{\text{pseudo}} = \frac{\partial \mathbf{F}(\boldsymbol{x})}{\partial \alpha_s} - \frac{\partial \mathbf{K}(\boldsymbol{x})}{\partial \alpha_s}\mathscr{U}(\boldsymbol{x}) \tag{7.23}$$

通常称为*伪载荷*. 为了计算伪载荷 $\mathbf{f}^{\text{pseudo}}$, 首先需要计算 $\partial\mathbf{K}(\boldsymbol{x})/\partial\alpha_s$ 和 $\partial\mathbf{F}(\boldsymbol{x})/\partial\alpha_s$. 为了叙述的流畅性, 将在下一小节中处理这一问题, 这里先假设它们是已知的.

将式 (7.22) 代入式 (7.20), 我们得到

$$\frac{\partial \hat{\Phi}(\boldsymbol{x})}{\partial \alpha_s} = \frac{\partial \Phi}{\partial \alpha_s} + \frac{\partial \Phi}{\partial \mathscr{U}}\mathbf{K}(\boldsymbol{x})^{-1}\left(\frac{\partial \mathbf{F}(\boldsymbol{x})}{\partial \alpha_s} - \frac{\partial \mathbf{K}(\boldsymbol{x})}{\partial \alpha_s}\mathscr{U}(\boldsymbol{x})\right). \tag{7.24}$$

在结构形状优化中, 有两种常见的方式来处理上式 (7.24), 即直接法和伴随分析方法. 在直接法中, 首先求解线性方程组 (7.22), 然后将得到的 $\partial\mathscr{U}(\boldsymbol{x})/\partial\alpha_s$ 代入到 (7.24) 中. 与之不同, 在伴随分析方法中, 首先解决如下的伴随问题

$$\mathbf{K}(\boldsymbol{x})\boldsymbol{\lambda}_s = \left(\frac{\partial \Phi}{\partial \mathscr{U}}\right)^{\mathrm{T}}. \tag{7.25}$$

然后将得到的 $\boldsymbol{\lambda}_s$ 代入到式 (7.24) 中.

当然, 两种方法计算出的 $\partial\hat{\Phi}(\boldsymbol{x})/\partial\alpha_s$ 结果相同, 但计算成本却截然不同. 乍一看, 我们的优化问题 (7.12) 只有一个目标函数而且没有约束, 伴随解析法似乎更为高效. 然而, 目标函数 (7.12a) 通常是通过数值积分 (例如 Gauss 数值积分) 计算得到的. 更具体地说, 目标函数值由以下公式计算得出

$$\hat{\Phi}(\boldsymbol{x}) \simeq \begin{cases} \sum\limits_{i=1}^{n_{gp}} \omega_{i,gp}\, \mathcal{C}_{abs}(\xi_{gp}, \eta_{gp}) & \text{若 } |\boldsymbol{\mathcal{J}}| > 0,\ \forall\,(\xi,\eta) \in \hat{\Omega}, \\ +\infty & \text{其他}. \end{cases} \tag{7.26}$$

其中 n_{gp} 是 Gauss 积分点 (ξ_{gp}, η_{gp}) 的总数量, $\omega_{i,gp}$ 是 Gauss 积分点所对应的权重.

在直接法中, 我们需要求解与设计变量个数相同的线性方程组 (7.22), 然后将结果代入到 (7.24) 中 n_{gp} 次. 在伴随法中, 需要先求解 (7.25)n_{gp} 次, 然后将结果代入到 (7.24) 中 n_{desg} 次, 其中 n_{desg} 是优化变量的个数. 通常, Gauss 点的总数 n_{gp} 等于 $(p+1)\cdot(q+1)\cdot \#ElemsU \cdot \#ElemsV$, 它的值远远大于 n_{desg}. 相比于代入和矩阵乘法运算, 求解线性方程组的计算成本要高得多. 因此, 为了更高的计算效率, 在我们的实现中采用直接的解析方法.

下面我们用一个撇号 $(\ )'$ 表示 $\partial/\partial\alpha_s$. 由于基函数与优化变量无关, 我们有

$$\begin{aligned}\boldsymbol{\mathcal{S}}'_{\xi} &= \mathbf{R}^{\mathrm{T}}_{A,\xi}\widetilde{\mathbf{Q}}', \quad \boldsymbol{\mathcal{S}}'_{\eta} = \mathbf{R}^{\mathrm{T}}_{A,\eta}\widetilde{\mathbf{Q}}', \quad \boldsymbol{\mathcal{S}}'_{\xi\xi} = \mathbf{R}^{\mathrm{T}}_{A,\xi\xi}\widetilde{\mathbf{Q}}', \\ \boldsymbol{\mathcal{S}}'_{\xi\eta} &= \mathbf{R}^{\mathrm{T}}_{A,\xi\eta}\widetilde{\mathbf{Q}}', \quad \boldsymbol{\mathcal{S}}'_{\eta\eta} = \mathbf{R}^{\mathrm{T}}_{A,\eta\eta}\widetilde{\mathbf{Q}}',\end{aligned} \tag{7.27}$$

其中 $\widetilde{\mathbf{Q}}' = [\mathbf{X}', \mathbf{Y}', \mathscr{U}']$, $\mathbf{X}'$ 和 $\mathbf{Y}'$ 是控制顶点 $\mathbf{Q}$ 的 x-分量和 y-分量的导数.

利用内积的导数性质和链式法则, 我们有

$$\begin{aligned}E' &= 2\langle \boldsymbol{\mathcal{S}}'_{\xi},\ \boldsymbol{\mathcal{S}}_{\xi}\rangle, \quad G' = 2\langle \boldsymbol{\mathcal{S}}'_{\eta},\ \boldsymbol{\mathcal{S}}_{\eta}\rangle, \\ F' &= \langle \boldsymbol{\mathcal{S}}'_{\xi},\ \boldsymbol{\mathcal{S}}_{\eta}\rangle + \langle \boldsymbol{\mathcal{S}}_{\xi},\ \boldsymbol{\mathcal{S}}'_{\eta}\rangle.\end{aligned} \tag{7.28}$$

而式 (7.4) 中单位法向量 $\mathbf{n}$ 的导数可以通过以下方式计算

$$\mathbf{n}' = \frac{(\boldsymbol{\mathcal{S}}_{\xi}\times\boldsymbol{\mathcal{S}}_{\eta})'}{\|\boldsymbol{\mathcal{S}}_{\xi}\times\boldsymbol{\mathcal{S}}_{\eta}\|} - \frac{\langle(\boldsymbol{\mathcal{S}}_{\xi}\times\boldsymbol{\mathcal{S}}_{\eta})', \boldsymbol{\mathcal{S}}_{\xi}\times\boldsymbol{\mathcal{S}}_{\eta}\rangle}{\|\boldsymbol{\mathcal{S}}_{\xi}\times\boldsymbol{\mathcal{S}}_{\eta}\|^2}\mathbf{n}, \tag{7.29}$$

其中 $(\boldsymbol{\mathcal{S}}_{\xi}\times\boldsymbol{\mathcal{S}}_{\eta})' = \boldsymbol{\mathcal{S}}'_{\xi}\times\boldsymbol{\mathcal{S}}_{\eta} + \boldsymbol{\mathcal{S}}_{\xi}\times\boldsymbol{\mathcal{S}}'_{\eta}$. 再次利用内积的导数性质和链式法则, 我们有

$$\begin{aligned}L' &= \langle \boldsymbol{\mathcal{S}}'_{\xi\xi}, \mathbf{n}\rangle + \langle \boldsymbol{\mathcal{S}}_{\xi\xi}, \mathbf{n}'\rangle, \quad M' = \langle \boldsymbol{\mathcal{S}}'_{\xi\eta}, \mathbf{n}\rangle + \langle \boldsymbol{\mathcal{S}}_{\xi\eta}, \mathbf{n}'\rangle, \\ N' &= \langle \boldsymbol{\mathcal{S}}'_{\eta\eta}, \mathbf{n}\rangle + \langle \boldsymbol{\mathcal{S}}_{\eta\eta}, \mathbf{n}'\rangle.\end{aligned} \tag{7.30}$$

接下来, 对 Gauss 曲率 (7.5) 和平均曲率 (7.6) 进行求导可得

$$K' = \frac{(LN-M^2)' - K\,(EG-F^2)'}{EG-F^2} \tag{7.31}$$

$$H' = \frac{1}{2}\frac{(LG-2MF+NE)' - 2H\,(EG-F^2)'}{EG-F^2}, \tag{7.32}$$

其中

$$\begin{aligned}\left(LN-M^2\right)' &= L'N + LN' - 2MM', \\ (LG-2MF+NE)' &= L'G + LG' - 2M'F - 2MF' + N'E + NE', \\ \left(EG-F^2\right)' &= E'G + EG' - 2FF'.\end{aligned} \tag{7.33}$$

逐点绝对主曲率 (7.10) 的导数为

$$\mathcal{C}'_{abs} = \begin{cases} 2H' & \text{若 } \kappa_{min} \geq 0, \\ \frac{1}{\sqrt{H^2-K}}(2HH'-K') & \text{若 } \kappa_{max} \geq 0, \\ -2H' & \text{其他}. \end{cases} \tag{7.34}$$

最终, 目标函数 (7.12a) 的导数为

$$\hat{g}'(\boldsymbol{x}) = \begin{cases} \int_{\hat{\Omega}} \mathcal{C}'_{abs}\, d\hat{\Omega} & \text{若 } |\boldsymbol{\mathcal{J}}| > 0,\ \text{对} \forall\, (\xi, \eta) \in \hat{\Omega}, \\ 0 & \text{其他}. \end{cases} \tag{7.35}$$

7.3.3 伪载荷的解析计算

现在, 我们需要计算式 (7.23) 中伪载荷 $\mathbf{f}^{\text{pseudo}}$ 的解析表达式. 为此, 我们首先考虑每个单元上拟载荷的贡献.

Algorithm 7.2: 计算目标函数值 Φ 及其关于设计变量的灵敏度 $grad\,\Phi$

Input: $\mathbf{x}_{desg}$: 优化变量组成的向量,
$activePts$: 优化变量的索引,
n_{desg}: 内部控制顶点的数量,
$\mathbf{P}$: 设计模型的控制顶点,
$\widetilde{\text{Refine}}$: 细化矩阵,
$\partial\mathbf{Q}/\partial\alpha_s$: 分析模型控制顶点关于设计变量的导数,
$\boldsymbol{\omega}_G$: Gauss 积分的权重,
$\mathbf{R}_A$, $\mathbf{R}_{A,\xi}$, $\mathbf{R}_{A,\eta}$, $\mathbf{R}_{A,\xi\xi}$, $\mathbf{R}_{A,\xi\eta}$, $\mathbf{R}_{A,\eta\eta}$: 分析模型基函数在 Gauss 积分点上的值和一阶二阶导数值;

Output: 目标函数值 Φ,
灵敏度 $grad\,\Phi$;

// 更新设计变量
$\mathbf{P}(activePts, :) = reshape(\mathbf{x}_{desg}, n_{desg}, 2)$; // 参数化
$\mathbf{Q} = \widetilde{\text{Refine}}\mathbf{P}$; // 灵敏度传播
// 下面的计算在分析模型上执行
计算单元刚度矩阵 $\mathbf{K}_e$, 单元载荷向量 $\mathbf{f}_e$, 并存储 Gauss 积分点处的物理坐标 $\boldsymbol{x}$ 和雅可比行列式的值 $|\boldsymbol{\mathcal{J}}|$ 等;
$grad\,\Phi = zeros(2\,n_{desg}, 1)$; // 初始化 $grad\ \Phi$ 为 $(n_{desg} \times 1)$ 向量
if 任一 *Gauss* 点上雅可比行列式 $|\boldsymbol{\mathcal{J}}|$ 小于零 **then**
  $\Phi = +\infty$;
  return;
end
组装全局刚度矩阵 $\mathbf{K}$ 和全局载荷向量 $\mathbf{f}$;
施加 Dirichlet 边界条件并求解离散平衡方程 (7.12b), 以计算 $\mathscr{U}$;
// 计算灵敏度
调用算法2 (目标函数值子程序)计算目标函数值 Φ;
// 计算灵敏度
调用算法2 (灵敏度分析子程序)计算灵敏度 $grad\,\Phi$;

首先, 对单元刚度矩阵求导, 我们得到

$$\mathbf{K}_e' = \int_{\hat{\Omega}_e} \left(\mathbf{G}'^{\mathrm{T}}\mathbf{G}\,|\mathcal{J}| + \mathbf{G}^{\mathrm{T}}\mathbf{G}'\,|\mathcal{J}| + \mathbf{G}^{\mathrm{T}}\mathbf{G}\,|\mathcal{J}|' \right)\, d\hat{\Omega}_e. \tag{7.36}$$

同理, 对单元载荷向量求导有

$$\mathbf{f}_e' = \int_{\hat{\Omega}_e} \mathbf{R}\left(\nabla_{\boldsymbol{x}} f \cdot \nabla_{\alpha_s}\boldsymbol{x}\,|\mathcal{J}| + f\,|\mathcal{J}|' \right)\, d\hat{\Omega}_e. \tag{7.37}$$

其中 $\nabla_{\boldsymbol{x}} f = [\partial f/\partial x,\ \partial f/\partial y]^{\mathrm{T}}$ 和 $\nabla_{\alpha_s}\boldsymbol{x} = [x',\ y']^{\mathrm{T}}$. 这里我们提醒读者, 矩阵 $\mathbf{G}$ 与分析模型的基函数有关, 而非设计模型.

Algorithm 2 (目标函数值子程序): 计算目标函数值 Φ

```
Φ = 0;                                                    // 初始化为 0
Q̃ = [Q, 𝒰];
for gp = 1; gp ≤ n_gp; gp++ do
    计算 S_ξ = R^T_{A,ξ} Q̃, S_η = R^T_{A,η} Q̃, S_ξξ = R^T_{A,ξξ} Q̃, S_ξη = R^T_{A,ξη} Q̃,
      S_ηη = R^T_{A,ηη} Q̃;
    计算 E, F, G;                                          // (7.2)
    计算单位法线向量 n;                                    // (7.4)
    计算 L, M, N;                                          // (7.3)
    计算 Gauss 曲率 K 和平均曲率 H;                        // (7.5), 7.6)
    计算最大主曲率 κ_max 和最小主曲率 κ_min;               // (7.7), (7.8)
    计算绝对主曲率 C_abs;                                  // (7.9)
    // 执行 Gauss 数值积分
    Φ = Φ + C_abs ω_{gp,G};
    存储 E, F, G, L, M, N 等后续需要使用的量;
end
```

式 (7.36) 和 (7.37) 中所涉及的余下的解析导数为

$$\begin{aligned} \mathbf{G}' &= -\mathbf{G}\mathbf{Q}'\mathbf{G}, \\ |\mathcal{J}|' &= |\mathcal{J}| \cdot\ trace(\mathbf{G}\mathbf{Q}'), \end{aligned} \tag{7.38}$$

其中 $trace(\,)$ 表示矩阵的迹. 感兴趣的读者可以参考文献[28]和文献[154]中的第 6.3.2.2 章节, 进一步了解式 (7.38) 中的两个表达式的推导过程.

类似于标准的等几何分析或有限元流程, 我们可以通过组装式 (7.36) 中单元刚度矩阵的导数来计算全局刚度矩阵的导数. 类似地, 我们也可以的到全局载荷向量 $\partial\mathbf{F}(\boldsymbol{x})/\partial\alpha_s$ 的导数. 一旦我们计算出伪载荷 $\mathbf{f}^{\mathrm{pseudo}}$, 那么 $\partial\mathscr{U}(\boldsymbol{x})/\partial\alpha_s$ 可以按照公式 (7.22) 计算得出.

Algorithm 2 (灵敏度分析子程序): Computation the sensitivity $grad\,\Phi$ w.r.t. the design variables α_s

```
// 为了符号上的清晰，我们把 ∂/∂α_s 记作 ( )′
for s = 1; s ≤ n_desg; s++ do
    计算单元刚度矩阵的导数 K′_e;                                   // (7.36)
    计算单元载荷向量的导数 f′_e;                                   // (7.37)
    组装伪载荷向量 f^pseudo;                                       // (7.23)
    施加 Dirichlet 边界条件并计算 ∂𝒰(x)/∂α_s;                      // (7.22)
    计算 ∂Q̃/∂α_s = [∂Q/∂α_s, ∂𝒰/∂α_s];                            // 即 Q̃′
    for gp = 1; gp ≤ n_gp; gp++ do
        计算 S′_ξ, S′_η, S′_ξξ, S′_ξη, S′_ηη;                      // (7.27)
        计算 E′, F′, G′;                                           // (7.28)
        计算单位外法向量的导数 n′;                                 // (7.29)
        计算 L′, M′, N′;                                           // (7.30)
        计算平均曲率的导数 H′;                                     // (7.32)
        // 执行 Gauss 数值积分.
        // 这里，gradΦ(s) 表示 gradΦ 的第 s 个分量.
        if κ_min,gp ≥ 0 then
            gradΦ(s) = gradΦ(s) + 2 H′ ω_gp,G;                     // (7.34), (7.35)
        end
        else if κ_max,gp ≥ 0 then
            计算 Gauss 曲率的导数 K′;                              // (7.31)
            gradΦ(s) = gradΦ(s) + (2 H H′ − K′)/√(H² − K) ω_gp,G;
        end
        else
            gradΦ(s) = gradΦ(s) − 2 H′ ω_gp,G;                     // (7.34), (7.35)
        end
    end
end
```

此时, 所有进行灵敏度分析所需的材料已经准备好. 为了提高本章算法的清晰度, 我们在算法7.2中详细总结和归纳优化问题 (7.12) 及其灵敏度分析的流程.

7.4 数值实验与对比分析

7.4.1 算法实现细节

我们在一台配备有 16GB 内存的 Intel Core i7-7700HQ CPU 的笔记本电脑上使用 MATLAB® 2020b 实现本章所提出的算法. 为了求解优化问题, 采用带有强 Wolfe 准则回溯线搜索的标准 L-BFGS 求解器. 关于该算法的详细解释, 可以参考文献[124]和文献[155]中的算法 7.5. 算法中所涉及到的积分全部采用 Gauss 数值积分进行计算, 线性方程组由 MATLAB 内置的反除命令求解.

在实验中, 我们采用了 h 细化和 p 细化方法获得分析模型. 在 h 细化中, 我们在每个单元的中点插入新的节点值. 而在 p 细化中, 我们同时对两个参数方向升阶. $p=1$ 表示分析模型是通过对设计模型进行一次升阶操作的到的. 需要注意的是, 更细的分析模型并不一定会得到更小的数值误差. 只要分析模型能够为模型能够为 r 自适应提供足够的数值精度即可. 换言之, 过度的细化分析模型是不必要的, 而且会导致较高的计算成本. 为此, 我们将细化次数限制在 $p \in \{0,1\}$ 和 $h \in \{0,1\}$ 的范围内.

7.4.2 与现有方法对比的设定

在文献[49]中, 作者将内部控制顶点作为优化变量并通过极小化等几何分析数值解与精确解之间的 L_2 误差来实现 r 自适应参数化. 该方法假定问题的精确解已知, 对于绝大多数实际应用中的问题并非如此. 然而, 我们仍可以将该方法得到的参数化结果作为最优参数化的参考. 具体来说, 该方法将以下能量函数作为目标函数

$$\begin{aligned} E &= \int_{\Omega} (\mathscr{U}^{\mathrm{h}} - \mathscr{U}^{\text{exact}})^2 \, d\Omega \\ &= \int_{\hat{\Omega}} \left(\mathbf{R}^{\mathrm{T}} \mathscr{U} - \mathscr{U}^{\text{exact}}(\boldsymbol{x})\right)^2 |\boldsymbol{\mathcal{J}}| \; d\hat{\Omega}, \end{aligned} \tag{7.39}$$

其中 $\mathscr{U}^{\mathrm{h}}$ 是等几何分析解, $\mathscr{U}^{\text{exact}}$ 表示精确解.

类似于 7.3 节中的推导, 能量 (7.39) 的解析灵敏度分析如下

$$\begin{aligned} \frac{\partial E}{\partial \alpha_s} = &\int_{\hat{\Omega}} 2\left(\mathscr{U}^{\mathrm{h}} - \mathscr{U}^{\text{exact}}\right)\left(\mathbf{R}^{\mathrm{T}} \frac{\partial \mathscr{U}}{\partial \alpha_s} - \nabla_{\boldsymbol{x}} \mathscr{U}^{\text{exact}} \cdot \nabla_{\alpha_s} \boldsymbol{x}\right) |\boldsymbol{\mathcal{J}}| \, d\hat{\Omega} \\ &+ \int_{\hat{\Omega}} \left(\mathscr{U}^{\mathrm{h}} - \mathscr{U}^{\text{exact}}\right)^2 \frac{\partial |\boldsymbol{\mathcal{J}}|}{\partial \alpha_s} d\hat{\Omega}, \end{aligned} \tag{7.40}$$

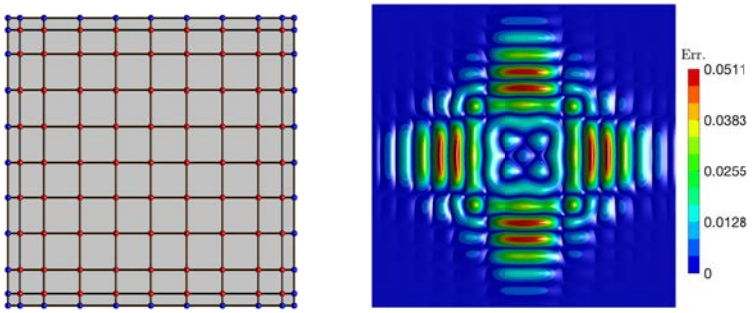
其中 $\nabla_{\boldsymbol{x}} \mathscr{U}^{\text{exact}} = [\partial \mathscr{U}^{\text{exact}}/\partial x, \; \partial \mathscr{U}^{\text{exact}}/\partial y]^{\mathrm{T}}$, $\nabla_{\alpha_s} \boldsymbol{x} = [\partial x/\partial \alpha_s, \; \partial y/\partial \alpha_s]^{\mathrm{T}}$.

此外, 我们还将与式 (7.13) 中的 Winslow 方法[104] 进行对比. 在这一方法中, 分析模型和设计模型具有相同的自由度, 因此难以平衡分析的精度和计算效率. 为此, 我们将本章提出的双层优化策略应用于该方法, 我们称这种方法为增强型 Winslow 方法以区别于原先的方法. 为了公平比较, 我们使用相同的 L-BFGS 方法求解器, 并且推导出其全解析的灵敏度分析.

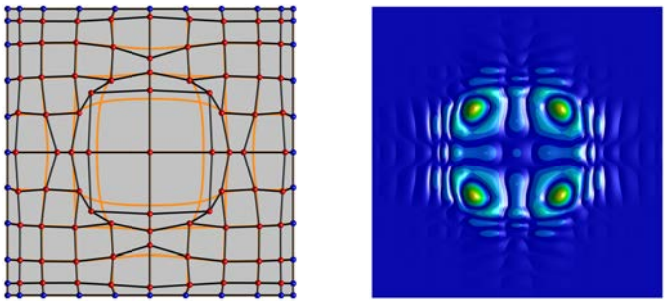
7.4.3 一个简单例子: 单位正方形

考虑在计算域 $\Omega = [0,1]^2$ 上的 Poisson 方程, 它具有如下的精确解

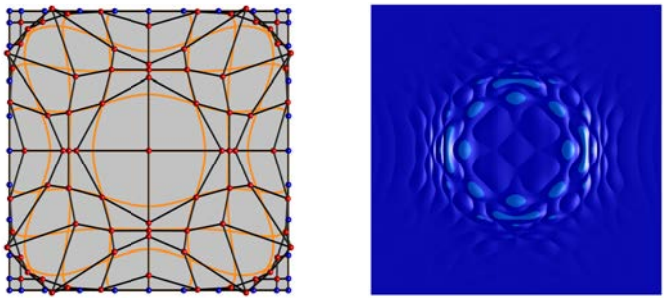
$$\mathscr{U}^{\text{exact}}(\boldsymbol{x}) = \tanh\left(\frac{0.25 - \sqrt{(x-0.5)^2 + (y-0.5)^2}}{0.05}\right) + 1. \tag{7.41}$$



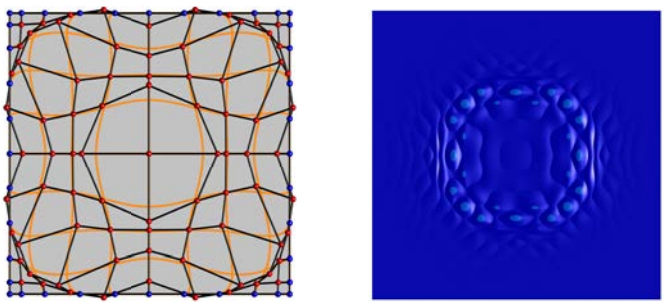
(a) 初始各向同性参数化, L_2 误差 = 1.2093e-02



(b) 分析模型未经细化得到, L_2 误差 = 4.4468e-03



(c) 分析模型经过一次 p 细化得到, L_2 误差 = 1.4674e-03



(d) 分析模型经过一次 h 细化得到, L_2 误差 = 1.1141e-03

图 7.3　单位正方形: 不同细化层次分析模型对数值误差的影响

Fig. 7.3　Unit Square: Impact of Different Refinement Levels on Analytical Models for Numerical Errors

如图 7.1 所示, 双线性 B 样条曲面足以表示正方形计算域. 不过此时由于没有内部控制顶点, 没有灵活性来调整参数化. 为此, 我们执行两次 p 细化, 然后进行三次 h 细

化, 来获得具有足够多的内部控制顶点以调整参数化. 如图 7.3(a) 所示, 设计模型具有 11×11 个控制顶点, 然后将内部 9×9 个控制顶点设置为优化变量以更新参数化. 之后, 我们通过 h 细化或 p 细化获得分析模型, 以确保获得充分的分析精度.

图 7.3 中左列展示了不同细化层次下得到的参数化结果. 可以看到, 得到的内部控制顶点和单元朝着数值误差较大的区域移动. 相应的误差图展示在右侧. 与初始的各向同性参数化结果相比, 所提出的方法明显地降低了等几何分析的数值误差. 随着细化程度的增加, 在相同自由度的情况下获得了更高的数值精度. 这是因为更准确的分析结果更好的反映了数值解的变化, 从而为自适应参数化提供了更好的指导. 这进一步印证了本章所提出的双层优化方法的有效性. 表 7.1 中数据表明, 与初始的各向同性参数化方法相比, 所提出的方法将最大误差, 平均误差和 L_2 误差减少了约 90%.

图 7.4 展示了 h 细化过程中的误差收敛曲线. 等几何分析的一个优势是在细化过程中, 计算域的参数化保持不变. 因此, 在得到 r 自适应的各向异性参数化结果后, 在更细层次的参数化上能够保持较高的数值精度. 换言之, 如果我们在较粗级别上获得了 r 自适应参数化结果, 那么在更细的级别上仍可以保持高精度优势.

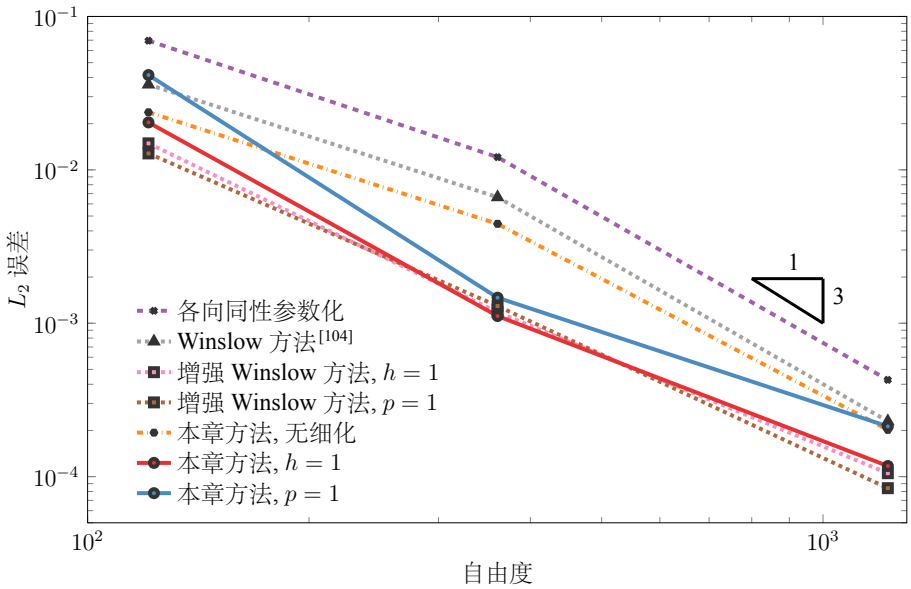


图 7.4　单位正方形: h 细化过程中误差收敛曲线

Fig. 7.4　Unit Square: Error Convergence Curves during h-Refinement

图 7.4 的结果还表明, 在细化级别为 0(即没有细化) 时, 我们的方法比原始 Winslow 方法[104] 表现更好. 这是因为 Winslow 项对降低数值误差的贡献有限, 从而影响参数化结果. 值得注意的是, 经过 p 或 h 细化的增强型 Winslow 方法的效果也得到了显著改善. 在本例中, 与我们的方法相比, 增强型 Winslow 方法在相同的细化级别下表现略好.

从计算效率的角度来看, 如表 7.1 所示, 本章方法显然优于 Winslow 方法. 为了保持参数化的双射性, 目标函数 (7.13) 中引入了 Winslow 项, 这在计算上是昂贵的, 尤其

表 7.1　不同 r 自适应参数化方法的量化统计数据
Tab. 7.1　Quantitative Data for Various r-Adaptive Methods

计算域	方法	细化层次	最大误差	平均误差	L_2 误差	时间 (秒)
单位正方形	各向同性	-	5.214e-02	6.480e-03	1.209e-02	0.28
	L_2 误差方法[49]	无细化	4.234e-03	6.478e-04	9.244e-04	26.39
	Winslow 方法[104]	无细化	3.361e-02	3.135e-03	6.630e-03	39.83
	增强 Winslow 方法	$p=1$	**6.125e-03**	**5.280e-04**	**1.180e-03**	77.37
		$h=1$	6.386e-03	6.085e-04	1.295e-03	247.61
	本章方法	无细化	3.866e-02	1.636e-03	4.447e-03	24.67
		$p=1$	7.234e-03	6.734e-04	1.467e-03	41.10
		$h=1$	**5.443e-03**	**4.901e-04**	**1.114e-03**	85.29
1/4 圆环	各向同性	-	1.369e-02	1.366e-03	4.352e-03	0.29
	L_2 误差方法[49]	无细化	1.405e-03	2.310e-04	5.533e-04	16.02
	Winslow 方法[104]	无细化	6.284e-03	5.396e-04	1.713e-03	19.62
	增强 Winslow 方法	$p=1$	**3.7651e-03**	**3.1402e-04**	**1.0606e-03**	41.19
		$h=1$	1.008e-02	6.649e-04	2.478e-03	81.92
	本章方法	无细化	7.139e-03	4.804e-04	2.098e-03	12.20
		$p=1$	4.078e-03	3.740e-04	1.350e-03	25.68
		$h=1$	**3.842e-03**	**3.512e-04**	**1.225e-03**	48.72
	本章方法 (加速后)	无细化	3.887e-03	3.938e-04	1.431e-03	4.70
		$p=1$	3.772e-03	4.516e-04	1.425e-03	7.94
		$h=1$	**3.369e-03**	**3.467e-04**	**1.130e-03**	15.79

是灵敏度分析. 相比之下, 本章方法仅计算绝对主曲率度量, 而避免了额外的计算.

7.4.4　NURBS 几何: 四分之一圆环

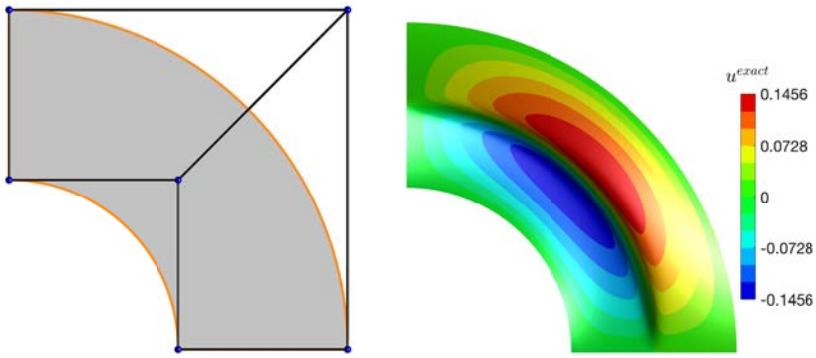
对于前一节所述的正方形计算域, B 样条表示已经足够. 然而, 对于具有圆锥曲线边界的计算域, 如椭圆或双曲线, NURBS 表示是必需的.

在本节中, 我们考虑在一个四分之一环形计算域上求解 Poisson 方程, 其精确解如下

$$\mathscr{U}^{\text{exact}}(\boldsymbol{x}) = \theta(\frac{\pi}{2} - \theta)(r-1)(r-2)\tanh\left(24\left(r-1.5\right)\right), \tag{7.42}$$

其中 $r = \sqrt{x^2+y^2}$, $\theta = atan(y/x)$. 精确解的图象如图 7.5(b) 所示.

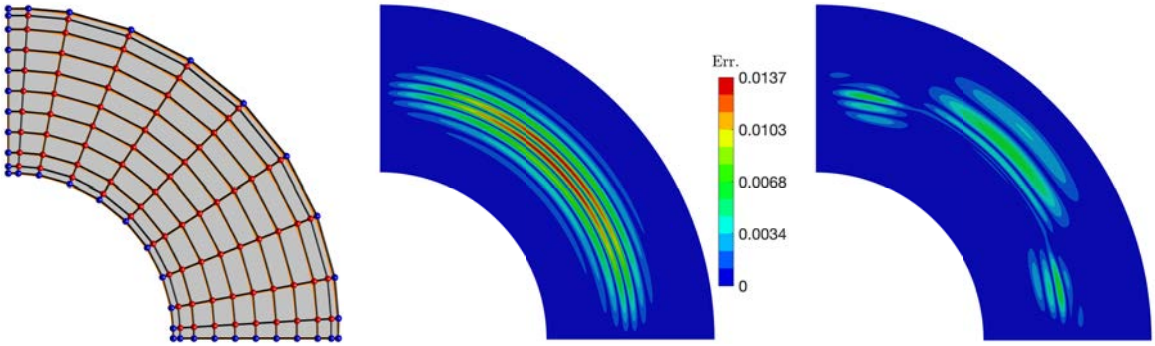
如图 7.5(a) 所示, 1×2 阶 NURBS 曲面足以表示一个四分之一环形. 我们在 u 参数方向上进行两次升阶, 在 v 参数方向上升阶一次, 随后进行三次 h 细化, 得到具有 11×11 个控制顶点的设计模型, 如图 7.6(a) 所示.



(a) 输入模型　　(b) 精确解

图 7.5　四分之一圆环: 输入模型和精确解

Fig. 7.5　Quarter Annulus: Input Model and Exact Solution



(a) 初始各向同性参数化　　(b) 初始 L_2 误差 = 4.3519e-03　　(c) 本章方法, $h = 0$, L_2 误差 = 2.0980e-03

(d) 本章方法, $h = 1$, L_2 误差 = 1.2247e-03　　(e) L_2 误差方法[49], L_2 误差 = 5.5329e-04　　(f) Winslow 方法[104], L_2 误差 = 1.7134e-03

图 7.6　四分之一圆环: 不同方法间的比较

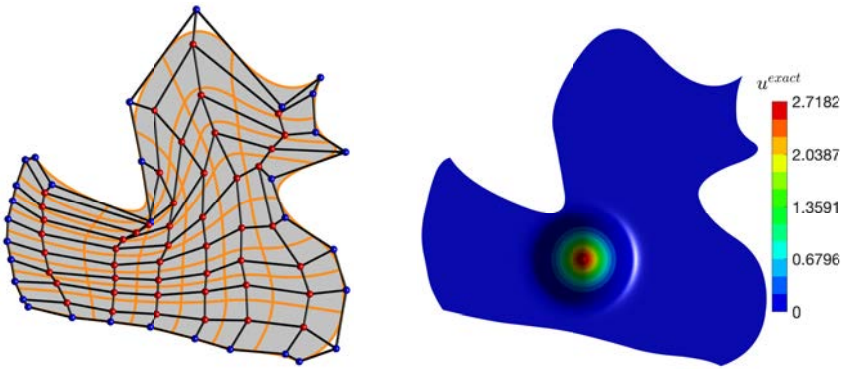
Fig. 7.6　Quarter Annulus: Comparison Among Different Methods

图 7.6 展示了各种方法所获得的数值误差. 可以看到, 所有 r 自适应方法都显著降低了数值误差. 图 7.6(c) 和图 7.6(d) 表明, 由于双层优化策略的使用, 本章的方法实现

了更高的数值精度. 虽然 L_2 误差法[49] 要优于其他两种方法, 但我们提醒读者, 在绝大多数实际应用中并没有已知的精确解. 表 7.1 详细展示了不同方法的量化统计.

7.4.5 更复杂的计算域几何

为了验证本章方法的鲁棒性, 我们进一步考虑一个具有更复杂几何形状的计算域. 具体来说, 我们考虑具有精确解 $\mathscr{U}^{\text{exact}}(\boldsymbol{x}) = \exp(1 - 300(x^2 + y^2))$ 的 Poisson 方程.



(a) 设计模型 (输入模型)　　(b) 精确解

图 7.7　鸭子: 设计模型和精确解
Fig. 7.7　Duck: Design Model and Exact Solution

图 7.7(a) 所示的输入模型是由本论文第 3 章方法所构造的各向同性参数化, 它是由双二次 B 样条曲面表示的. 在这个实例中, 我们将设计模型与输入模型设置为相同. 相应的精确解如图 7.7(b) 所示.

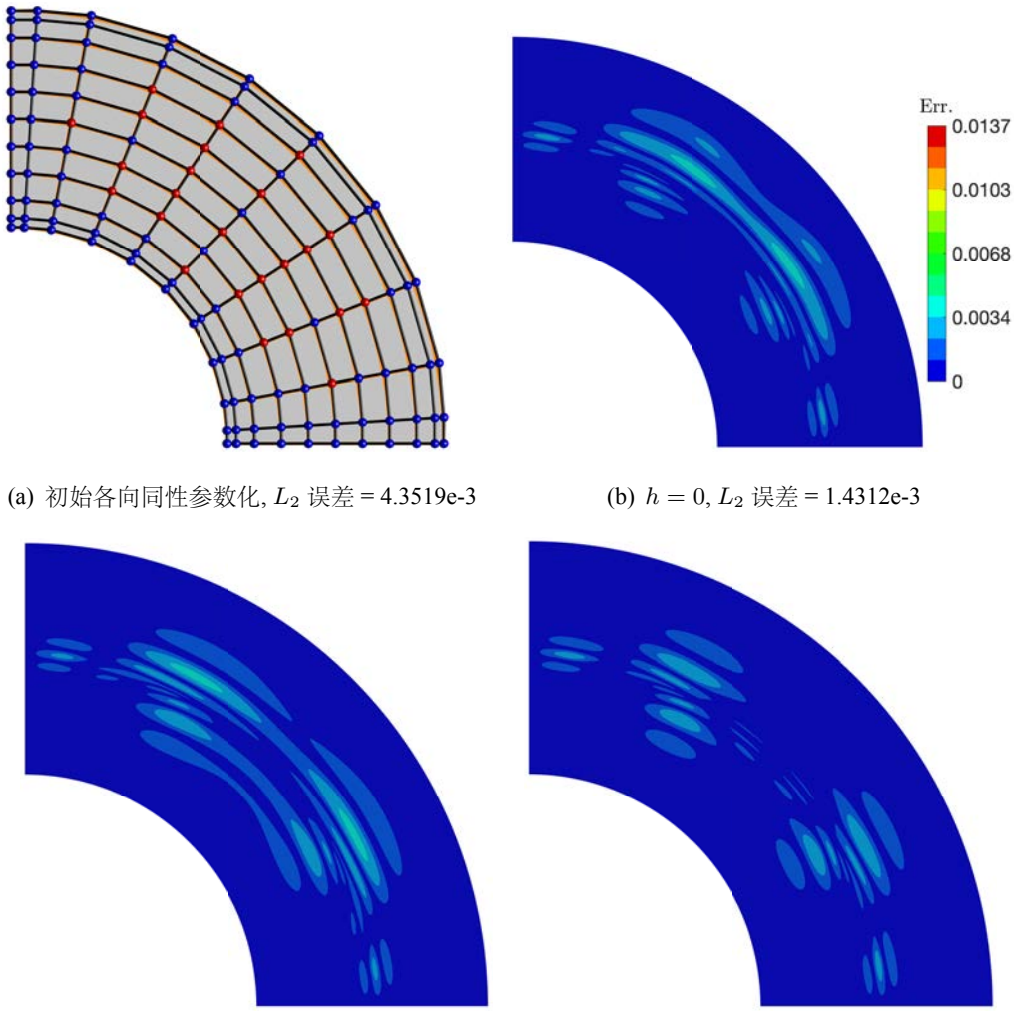
表 7.2　鸭子模型上不同方法的量化统计数据
Tab. 7.2　Quantitative Data for Duck Model by Different Methods

细化层次	平均误差	L_2 误差	计算时间 (秒)
各向同性参数化	1.5032e-02	2.7269e-02	0.21
无细化	6.2290e-03	1.1213e-02	6.03
$p = 1$	**2.1423e-03**	**4.5594e-03**	9.28
$h = 1$	2.4356e-03	7.0504e-03	15.60

如表 7.2 所示, 与各向同性参数化结果相比 (表 7.2 的第一行), 本章方法均有效地降低了计算误差. 此外, 若采用我们所提出的双层优化策略, 可以得到更令人满意的数值结果. 在这个例子中, p 自适应细化表现最佳.

7.4.6 加速策略

考虑到具有局部特征问题的特点, 将所有的内部控制顶点作为优化变量是不划算的, 特别是就计算成本而言. 如图 7.3(b) 所示, 远离解曲面变化较大区域的控制顶点对结果的影响很小, 甚至几乎保持不变. 基于这一观察, 本节中考虑一种简单的加速策略, 即只设置少量的内部控制顶点作为优化变量.



(a) 初始各向同性参数化, L_2 误差 = 4.3519e-3

(b) $h = 0$, L_2 误差 = 1.4312e-3

(c) $p = 1$, L_2 误差 = 1.4252e-3

(d) $h = 1$, L_2 误差 = 1.1296e-3

图 7.8 加速策略: 仅将内部红色的控制顶点作为优化变量

Fig. 7.8 Acceleration Strategies: Using Only Internal Red Control Points as Optimization Variables

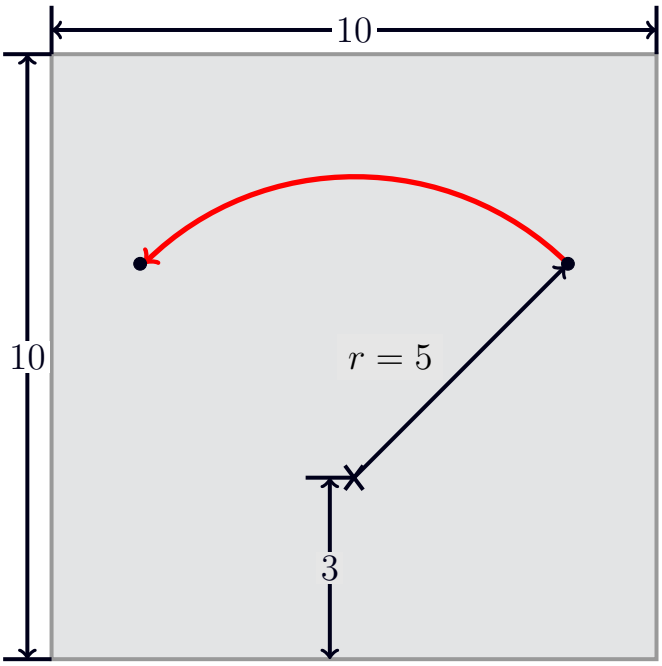
具体来说, 在迭代之前, 我们先计算所有内部控制顶点的灵敏度. 然后, 我们将那些 $(\partial\Phi/\partial x_s)^2 + (\partial\Phi/\partial y_s)^2$ 值较大的点设置为设计变量, 其中 $\partial\Phi/\partial x_s$ 和 $\partial\Phi/\partial y_s$ 分别是目标函数 Φ 关于第 s 个内部控制顶点的 x 和 y 分量的偏导数.

如图 7.8(a) 所示, 我们将 24 个控制顶点 (红色标记) 设置为优化变量. 图 7.8 中的误差分布说明了这一方法可以有效的降低计算误差. 在相同的终止准则下, 该策略的

计算性能远远优于将所有内部控制顶点设置为优化变量的情况. 尤为重要的是, 该策略在计算效率上获得了显著提升, 其速度提升了超过 50%. 请参见表 7.1.

7.5 在带有移动 Guassian 热源热传导问题上的应用

移动热源问题在焊接分析, 3D 增材制造[98] 等瞬态分析问题中十分常见. 为了展示本章方法的实用性, 我们考虑一个带有移动 Gaussian 热源的二维线性热传导问题, 如图 7.9 所示.



激光功率 P	9×10^5	[W]
激光扫描速度	1.57	[mm/sec]
吸收率 η	0.33	
源半径 r_h	100	$[\mu m]$
导热系数 κ	1.0	$[W/mm/K]$
热容量 C_p	1.0	$[J/kg/K]$
密度 ρ	1.0	$[kg/mm^3]$
初始温度 $\mathscr{U}_0$	20.0	$[^\circ C]$

图 7.9 移动热源问题描述: 热源沿着圆弧移动, 红色箭头表示扫描路径
Fig. 7.9 Problem Description for Moving Heat Source: Heat Source Travels Along an Arc, Red Arrows Indicate Scanning Path

具体来说, 我们考虑在时域 $T = [0, 1]$ 和空间域 $\Omega = [0, 1]^2$ 上的以下初边值问题

$$C_p\rho\frac{\partial u(\boldsymbol{x}, t)}{\partial t} - \nabla \cdot (\kappa\nabla\mathscr{U}(\boldsymbol{x}, t)) = f(\boldsymbol{x}, t) \qquad in\ \Omega \times T, \tag{7.43a}$$

$$\mathscr{U}(\boldsymbol{x}, t) = \mathscr{U}_0, \qquad in\ \Omega, \tag{7.43b}$$

$$\kappa\nabla\mathscr{U}(\boldsymbol{x}, t) = 0, \qquad on\ \partial\Omega \times T, \tag{7.43c}$$

其中 $\mathscr{U}(\boldsymbol{x}, t)$ 是未知的温度场, $f(\boldsymbol{x}, t) = P\eta\ \exp\left(-\left((x - x_0)^2 + (y - y_0)^2\right)/r_h^2\right)$ 是外部的 Gaussian 热源,

$$\begin{cases} x_0 = 5\cos(\frac{\pi}{2}t + \frac{\pi}{4}) + 5 \\ y_0 = 5\sin(\frac{\pi}{2}t + \frac{\pi}{4}) + 3 \end{cases} \tag{7.44}$$

是热源的扫描轨迹, 式 (7.43b) 表示初值条件, 式 (7.43c) 是绝热 Neumann 边界条件, $\mathbf{n}$ 是单位外法线向量, 其余参数如图 7.9 所示.

在时刻 t, 根据虚温原理和等几何离散化, 问题 (7.43) 的离散平衡方程为

$$\mathbf{M}_{(t)}\frac{\partial \mathscr{U}_{(t)}}{\partial t}+\mathbf{K}_{(t)}\mathscr{U}_{(t)}=\mathbf{f}_{(t)}, \tag{7.45}$$

其中 $\mathbf{M}_{(t)}=C_p\rho\int_\Omega \mathbf{R}^{\mathrm{T}}\mathbf{R}d\Omega$ 是时刻 t 的质量矩阵, $\mathbf{K}_{(t)}=\kappa\int_\Omega \mathbf{G}_{(t)}^{\mathrm{T}}\mathbf{G}_{(t)}d\Omega$ 表示时刻 t 的刚度矩阵, $\mathbf{f}_{(t)}=\int_\Omega \mathbf{R}f(\boldsymbol{x},t)d\Omega$ 是时刻 t 的载荷向量.

接着, 使用向后欧拉方法对离散的平衡方程 (7.45) 进行时间上的离散. 在 $t+\Delta t$ 时刻, 离散平衡方程可以表示为

$$\left(\mathbf{M}_{(t)}+\Delta t\mathbf{K}_{(t)}\right)\mathscr{U}_{(t+\Delta t)}=\Delta t\mathbf{f}_{(t+\Delta t)}+\mathbf{M}_{(t)}\mathscr{U}_{(t)}, \tag{7.46}$$

其中 Δt 是常数时间增量, $\mathscr{U}_{(t)}$ 表示时刻 t 的解向量. 实现中, 我们设置 $\Delta t=0.05$.

由于该瞬态平衡方程涉及到质量矩阵, 其灵敏度分析与前面的稳态 Poisson 方程稍有不同. 在每个时刻 $t+\Delta t$, 我们按照如下的公式计算分析模型的灵敏度

$$\frac{\partial \mathscr{U}_{(t+\Delta t)}}{\partial \alpha_s}=(\mathbf{M}_{(t)}+\Delta t\mathbf{K}_{(t)})^{-1}\mathbf{f}_{(t)}^{\text{pseudo}}, \tag{7.47}$$

其中

$$\mathbf{f}_{(t)}^{\text{pseudo}}=\Delta t\left(\frac{\partial \mathbf{f}_{(t+\Delta t)}}{\alpha_s}-\frac{\partial \mathbf{K}_{(t)}}{\partial \alpha_s}\mathscr{U}_{(t+\Delta t)}\right)+\frac{\partial \mathbf{M}_{(t)}}{\partial \alpha_s}\left(\mathscr{U}_{(t)}-\mathscr{U}_{(t+\Delta t)}\right) \tag{7.48}$$

是时刻 t 的伪载荷向量. 上式 (7.48) 中其余项 $\partial\mathscr{U}_{(t)}/\alpha_s$, $\partial\mathbf{K}_{(t)}/\partial\alpha_s$ 和 $\partial\mathbf{f}_{(t+\Delta t)}/\alpha_s$ 的计算方式分别与 (7.22), (7.36) 和 (7.37) 是一致的.

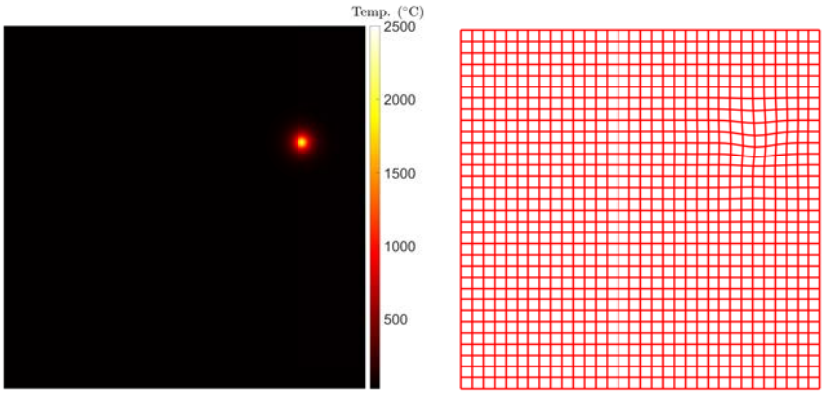
对单元质量矩阵 $\mathbf{M}_{(t),e}$ 进行微分, 我们可以得到

$$\frac{\partial \mathbf{M}_{(t),e}}{\partial \alpha_s}=\int_{\hat{\Omega}_e}\mathbf{R}^{\mathrm{T}}\mathbf{R}\frac{\partial\left|\boldsymbol{\mathcal{J}}\right|_{(t)}}{\partial \alpha_s}\,d\hat{\Omega}_e. \tag{7.49}$$

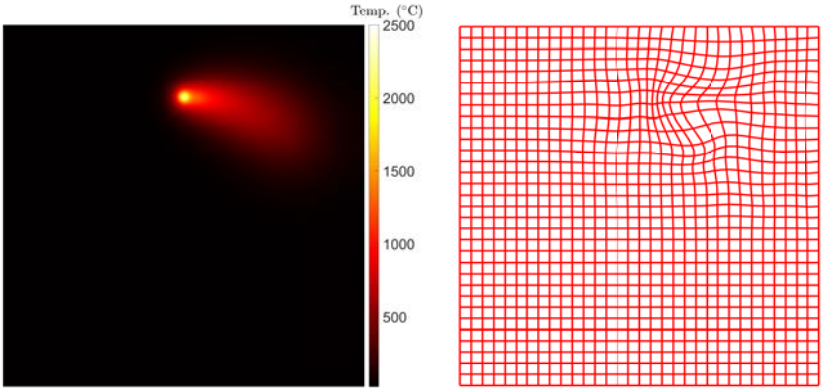
随后, 利用与刚度矩阵组装一样的方式, 我们可以得到 $\partial\mathbf{M}(t)/\partial\alpha_s$ 的表达式. 一旦计算出 $\partial\mathscr{U}(t+\Delta t)/\partial\alpha_s$, 那么每个时刻 $t+\Delta t$ 的灵敏度分析与稳态 Poisson 方程完全一致.

对于问题 (7.43), 很难得到其解析解, 因此我们使用具有 265225 自由度的双三次 B 样条计算参考解 $\mathscr{U}_{(t),ref}$. 根据文献[98], 我们定义时刻 t 的系统的总能量

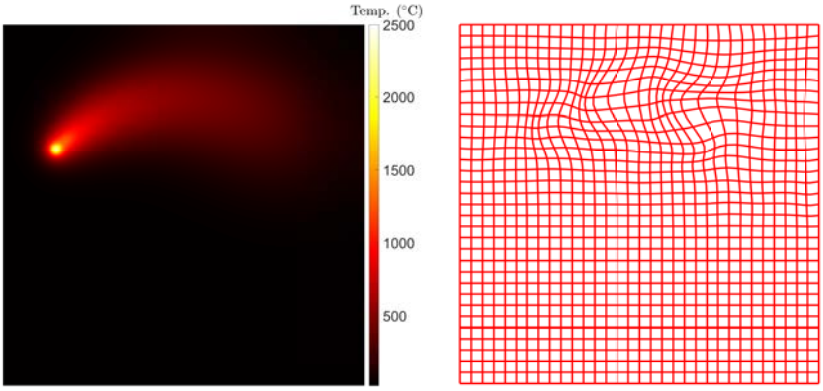
$$\mathcal{E}(\mathscr{U}_{(t)},\mathscr{U}_{(t)})=\frac{1}{2}\left(\kappa\int_\Omega\nabla\mathscr{U}_{(t)}\cdot\nabla\mathscr{U}_{(t)}d\Omega+C_p\rho\int_\Omega\mathscr{U}_{(t)}\frac{\partial\mathscr{U}_{(t)}}{\partial t}d\Omega\right).$$



(a) $t = 0.05$ 秒



(b) $t = 0.50$ 秒



(c) $t = 1.00$ 秒

图 7.10　不同时刻的解和参数化结果

Fig. 7.10　Solutions and Parameterization Results at Different Time Instances

为了度量数值误差, 我们采用以下相对总能量误差

$$\varepsilon(\mathscr{U}_{(t)}, \mathscr{U}_{(t)}) = \sqrt{\frac{\left|\mathcal{E}(\mathscr{U}_{(t),ref}, \mathscr{U}_{(t),ref}) - \mathcal{E}(\mathscr{U}_{(t)}, \mathscr{U}_{(t)})\right|^2}{\left|\mathcal{E}(\mathscr{U}_{(t),ref}, \mathscr{U}_{(t),ref})\right|^2}}.$$

图7.10展示了不同时刻各向异性的参数化结果. 可以看到, 等参线集中到数值解变化较大的区域. 不同时刻下的相对总能量误差如图 7.11 所示. 与各向同性的参数化相比, 本章所提出的方法在所有的时刻均有效降低了数值误差.

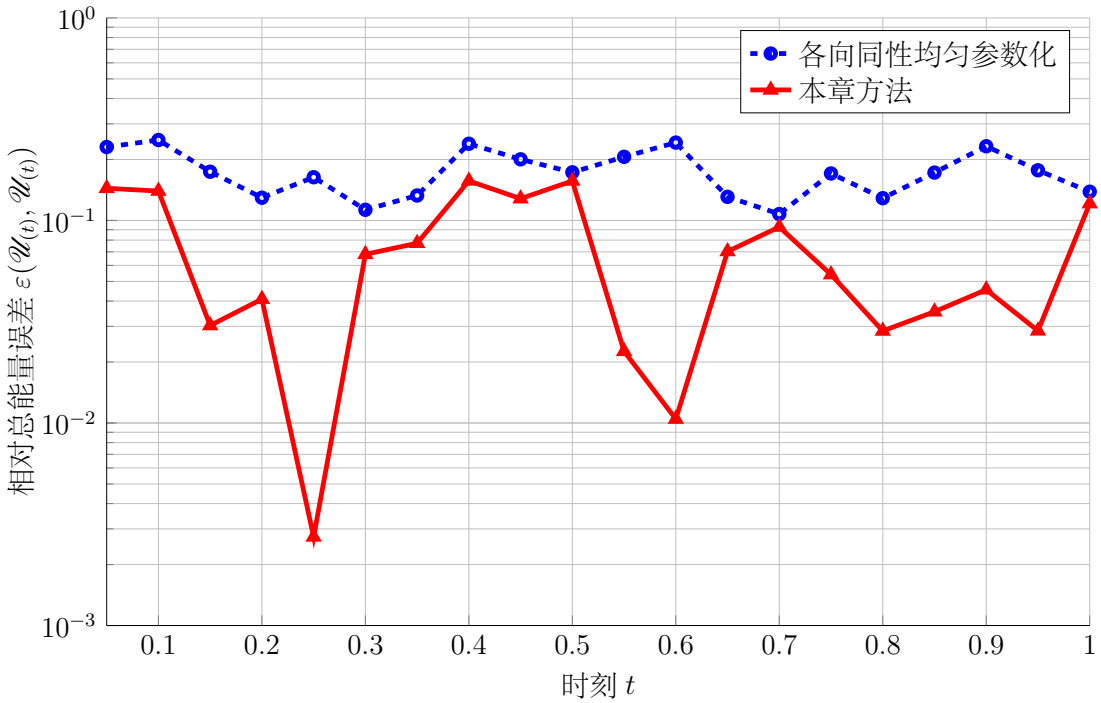


图 7.11　不同时刻 t 的相对总能量误差 $\varepsilon(\mathscr{U}_{(t)}, \mathscr{U}_{(t)})$, 此时自由度为 1225

Fig. 7.11　Relative total energy $\varepsilon(\mathscr{U}_{(t)}, \mathscr{U}_{(t)})$ at Different Time Instances t, with DOFs = 1225

7.6　本章小结

本章提出了一种基于绝对主曲率度量的等几何分析 r 自适应参数化方法. 在保持自由度相同的前提下, 此方法能实现更高精度的各向异性参数化. 为了确保参数化结果的双射性, 我们对目标函数进行了正则化处理. 为了提高计算效率和数值精度, 采取了双层优化策略, 并详细推导了设计模型至分析模型的灵敏度传播与分析公式. 为了尽可能清晰且详细地阐述灵敏度传播和灵敏度分析过程, 提供了详尽的算法描述. 通过多个实际计算实例和对比实验, 验证了该方法的有效性. 最终将该方法成功应用于移动热源问题中, 数值结果证明了该方法能够有效降低计算误差.

本章所提出的方法仍然存在一些不足之处. 首先, 该方法无法直接推广至 $\mathbb{R}^3$ 中的壳或体问题上, 这是因为在 $\mathbb{R}^4$ 空间中缺乏曲率的等价概念. 为解决这一问题, 可以考虑使用后验误差估计. 随后, 本章给出的灵敏度分析和灵敏度传播方法可以几乎无缝地应用于这些方法中. 第二个局限性是, 虽然本章探讨了加速策略, 但是数值结果表明, 当前的方法在计算时间上仍难以超越使用少量全局 h 细化的方法. 一种可能的改进方向是将本章的方法与具有局部细化能力的样条技术 (如 T 样条或 THB 样条) 相结合.

8 总结与展望

8.1 本文工作总结

本论文聚焦于等几何分析中高质量高效适合分析的参数化构造方法: 其中第3章至第5章考虑了各向同性参数化方法, 分别是障碍函数方法、惩罚函数方法、和椭圆参数化方法; 第6章将所提出的方法应用到真实工业场景中双螺杆压缩机的网格生成问题中; 而第7章则专注于各向异性参数化方法. 除了方法和理论贡献外, 还着眼于所提出方法的实际应用价值. 总结来说, 本文的研究工作包括以下几个方面:

在第3章中, 我们首先讨论了一般平面 NURBS 参数化单值性的充分条件和必要条件, 并且设计了一个实用的单值性检验算法. 尽管控制点和权因子都会影响参数化效果, 但目前还没有相关工作考虑权因子对参数化质量的影响. 我们引入内部控制顶点和权因子作为优化变量, 以提高计算域参数化的质量. 该方法避免了计算和更新参数化雅可比行列式的系数, 并将消除折叠和参数化质量提升分离开, 实现了一种基于障碍函数的参数化方法. 与现有方法相比, 本章提出的方法在计算效率和参数化质量方面表现出更好的性能. 通过考虑权因子的优化, 为具有复杂几何形状的计算域提供了一种更高效且更高质量的参数化方法.

在第4章中, 我们提出了一种基于罚函数的体参数化方法, 以解决第3章中障碍函数方法需要额外消除折叠步的局限性, 并推广至三维区域参数化问题. 利用散度定理, 仅通过模型边界表示计算物理域的体积, 进而构造出刻画参数化体积畸变的目标泛函项. 提出了一种新的惩罚函数, 能够有效降低误差. 结合雅可比正则化思想, 算法可以直接接受有自相交的初始参数化. 为提高基于梯度的优化过程中的稳定性, 推导出了完全解析的梯度计算公式. 为提高所提出方法的计算效率, 我们提议使用减缩数值积分策略. 数值试验结果表明, 本章参数化方法展现出了更高的计算效率和鲁棒性.

在第5章中, 针对具有极端纵横比的狭长计算域, 提出了一种改进的基于偏微分方程的椭圆参数化方法. 为削弱椭圆网格生成方法中单元尺寸不均匀的影响, 在 Sobolev 空间 H^1 中提出了一种缩放版本的新离散方法. 为提高计算效率和数值稳定性, 推导出了 H^2 和 H^1 离散的解析雅可比矩阵. 此外, 为了加速求解所涉及的非线性方程组, 我们首先将该问题转化为固定点迭代问题, 随后提出预条件 Anderson 加速算法来加速其收敛. 数值实验表明本章方法大大提升了算法的数值稳定性和计算效率.

在第6章中, 将第5章所提出的方法应用到真实的工业应用场景, 即双螺杆压缩机的网格生成问题中. 利用椭圆参数化方法构造一系列平面参数化, 随后利用放样技术构造出高阶体样条参数化结果. 利用不同的网格离散方案, 能够轻易地生成具有边界层特性的网格, 还可以实现与流场方向一致的结构六面体网格. 在商用软件 ANSYS

CFX™ 中的仿真测试表明, 本方法生成的网格可以适用于复杂的流体力学仿真分析.

在第7章中, 针对具有局部特征的物理问题, 提出了一种基于曲率度量的各向异性参数化方法. 该方法的基本思路是将 IGA 数值解视为 $\mathbb{R}^3$ 中的参数曲面, 并利用绝对主曲率来刻画 IGA 解的变化, 从而建立几何量和分析结果之间的紧密联系. 为避免自相交, 修正了目标函数, 并借助于带有线搜索准则的优化算法, 有效地保证了参数化结果的单值性. 为提高基于梯度的优化算法的计算效率和数值稳定性, 推导出灵敏度传播和解析的灵敏度计算公式. 采用具有相同几何形状而细化程度不同的双层方法, 在粗参数化 (设计模型) 层次上更新参数化以提高计算效率, 在细参数化 (分析模型) 层次上执行等几何仿真分析以提高模拟精度. 展示了从设计模型到分析模型的灵敏度传播以及解析灵敏度计算的详细算法. 与各向同性参数化相比, 本章方法在许多情形下能够显著提高计算效率和数值精度. 同时, 所提出的方法在处理具有复杂几何特征和物理特性的问题时表现出良好的鲁棒性. 最后, 将所提出的方法应用于具有移动高斯热源的二维线性热传导问题, 这是增材制造应用中的一个简化模型.

8.2 未来工作展望

(1) 多块体参数化

实际工程中的几何结构往往十分复杂, 如剪裁曲面或高亏格情形. 处理这些复杂几何结构的参数化通常十分具有挑战性. 为解决这一问题, 往往需要首先将复杂计算域分割为简单、易于处理的六面体块, 如图8.1所示. 对于等几何分析应用, 如何在确保边界表示精度的同时, 高效生成简单的六面体分块参数化是一个亟需解决的问题. 这不仅需要设计专门的算法, 而且可能涉及到多种技术, 如局部参数化、拓扑优化和简化等. 开发一种能够自动生成六面体拓扑结构的有效方法将为等几何分析提供一个强有力的工具, 极大地促进等几何分析的应用和发展.

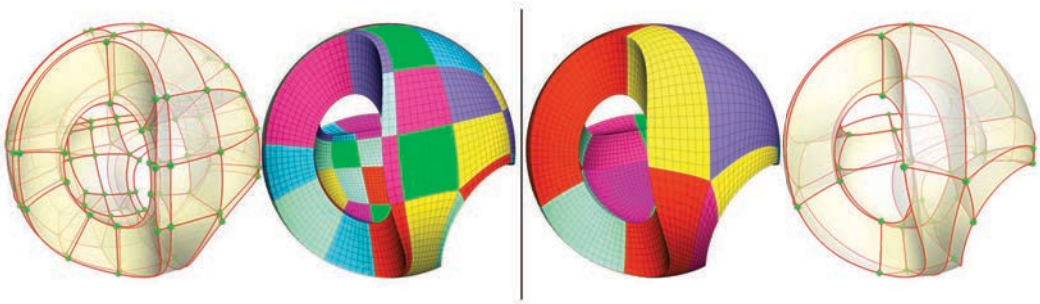


图 8.1 六面体块分割[156]
Fig. 8.1 Hexahedral Block Partitioning[156]

(2) 处理非标准 NURBS 几何

除了处理传统的 NURBS 参数化问题, 具有局部细化能力的样条工具在自适应等

几何分析中展现出了极大的潜力, 如 T 样条、THB 样条、PHT 样条等. 这些非标准样条在提高参数化灵活性和适应性方面具有显著优势, 但同时也带来了额外的计算复杂性和求解挑战. 研究基于这些新型样条的参数化方法对于提高等几何分析的适用性和性能至关重要. 特别是在捕捉计算域某些边界附近的细小特征时, 具有极大的潜力. 近年来, 虽然已有若干研究关注到这一领域, 如[65,87,157], 但在处理多块体参数化时, 如何在几何连续性约束下构建光滑且适合分析的参数化仍然是一个待解决的有趣问题[158]. 值得一提的是, 目前在 G+Smo 中的实现已经具备处理 THB 样条参数化的能力. 然而遗憾的是, 作者对参数化问题中局部细化策略的研究尚在起步阶段, 这部分将留作未来进一步探究的课题.

(3) 自适应参数化

物理问题通常天然具有多尺度特性. 以流体力学为例, 在超声速空气流动中的激波、多相流中的两相界面、以及几何曲率极大的区域等, 都具有典型的跨尺度特征. 为了在计算量可接受的同时实现高精度模拟, 对计算域进行精确且有针对性的参数化局部加密或者移动参数化单元是必不可少的. 在流体动力学模拟中, 边界层和激波区域常常是加密的焦点; 而在结构力学中, 应力集中区和断裂点则常常受到重点关注. 然而在实际模拟环境中, 往往难以预测跨尺度特征出现的区域和时机. 这就需要网格具备随物理场变化而自适应加密的能力, 同时也带来了关于网格加密规则的一系列问题, 以及在并行计算环境下如何实现动态负载均衡的性能挑战.

此外, 在实际工程应用中, 许多问题涉及多材料或多学科耦合, 如结构–流体耦合分析. 在这种情况下, 参数化问题需要同时考虑多个物理场的特性, 进一步增加了问题的复杂性和求解难度. 因此, 需要开发能够适应不同物理场需求的参数化方法和技术.

(4) 工业软件开发

为促进等几何分析在实际工程领域的广泛应用, 不仅需要理论和算法的突破, 还需研发用户友好、高效能的工业软件. 这需要在算法设计和软件工程两个方向上同步推进. 首先, 在用户界面方面, 应设计直观、易于操作的交互界面, 并配备一系列丰富的示例和教程, 以降低用户的学习曲线. 这不仅让工程师和研究人员能更容易地掌握该技术, 也有助于多种工程问题中的实际应用. 其次, 软件的兼容性和可扩展性也是关键. 理想的软件应能与现有的工程软件环境无缝集成, 并支持模块化配置, 以便用户根据特定需求进行灵活的功能扩展. 另外, 在算法和计算性能方面, 不断优化现有参数化方法是至关重要的. 这不仅涉及计算效率的问题, 还需要确保算法的准确性和稳定性. 为适应不断增长的工程应用复杂性, 探索和开发新的参数化技术和策略是不可或缺的. 此外, 考虑到不同工程领域具有各自的特性和挑战, 研发专门针对特定场景定制的参数化方法将极大地提升软件在各领域中的适用性和性能.

8.3 创新点摘要

- **提出了两类基于优化的高效参数化方法.** 提出基于障碍函数的三步参数化方法, 避免计算和更新参数化雅可比行列式的系数, 从而大幅提升了计算效率. 通过交替更新内部的控制顶点和权因子, 有效提高了具有复杂几何形状计算域的参数化质量. 提出了平面 NURBS 参数化单值性的充分条件和必要条件, 并设计了一种实用的单值性检验算法. 设计了基于罚函数的参数化方法, 引入了新的惩罚函数, 有效降低了计算误差. 利用散度定理, 仅通过模型的边界表示求出计算域的体积, 构造了刻画参数化体积扭曲的目标泛函项. 结合雅可比正则化思想, 该方法可以直接接受带有自相交的参数化作为初值, 进一步简化了计算流程.

- **提出了预条件 Anderson 加速求解器集成的改进椭圆参数化方法.** 针对具有极端纵横比的计算域, 提出了一种改进的椭圆参数化算法, 并给出了其在 Sobolev 空间 H^1 中的新离散格式, 有效提升了参数化质量. 推导出了所涉及非线性方程组的全解析雅可比矩阵计算公式. 引入了带有动态更新策略的预条件 Anderson 加速算法, 增强了数值稳定性和求解效率, 并给出该算法残差界的估计和收敛性分析. 该方法已成功应用于双螺杆压缩机流体力学仿真的真实工业场景中.

- **提出了基于曲率度量和双层优化策略的各向异性参数化方法.** 将 IGA 数值解视作 $\mathbb{R}^3$ 空间中的参数曲面, 利用绝对主曲率来表征 IGA 解的变化程度. 借助带有线搜索准则的优化算法, 修正后的目标函数可以有效保证参数化结果的单值性. 使用具有相同几何形状而细化程度不同的双层优化策略, 实现在粗参数化层次上更新参数化以提高计算效率, 并在细参数化层次上执行等几何分析以提高计算精度. 推导出了灵敏度传播和解析的灵敏度计算公式, 提高了基于梯度的优化方法中的数值稳定性. 相较于各向同性参数化, 本方法在处理具有局部特征的复杂物理问题时展现了良好的适用性和计算效率.

参 考 文 献

[1] Pan M-D, Chen F-L, Tong W-H. Low-rank parameterization of planar domains for isogeometric analysis [J]. Computer Aided Geometric Design, 2018, 63: 1–16.

[2] Pan M-D, Chen F-L, Tong W-H. Volumetric spline parameterization for isogeometric analysis [J]. Computer Methods in Applied Mechanics and Engineering, 2020, 359: 112769.

[3] Liu H, Yang Y, Liu Y, et al. Simultaneous interior and boundary optimization of volumetric domain parameterizations for IGA [J]. Computer Aided Geometric Design, 2020, 79: 101853.

[4] Farin G, Hoschek J, Kim M-S. Handbook of Computer Aided Geometric Design [M]. Elsevier, 2002.

[5] Sarcar M, Rao K M, Narayan K L. Computer Aided Design and Manufacturing [M]. PHI Learning Pvt. Ltd., 2008.

[6] Schoenberg I J. Contributions to the problem of approximation of equidistant data by analytic functions. Part B. On the problem of osculatory interpolation. A second class of analytic approximation formulae [J]. Quarterly of Applied Mathematics, 1946, 4 (2): 112–141.

[7] Farin G E, Farin G. Curves and Surfaces for CAGD: A Practical Guide [M]. Morgan Kaufmann, 2002.

[8] Piegl L, Tiller W. The NURBS Book [M]. Springer Science & Business Media, 1996.

[9] Courant R. Variational methods for the solution of problems of equilibrium and vibrations [J]. Bulletin of the American Mathematical Society, 1943, 49 (1): 1–23.

[10] Hughes T J R. The Finite Element Method: Linear Static and Dynamic Finite Element Analysis [M]. Courier Corporation, 2012.

[11] Cottrell J A, Hughes T J R, Bazilevs Y. Isogeometric Analysis: Toward Integration of CAD and FEA [M]. John Wiley & Sons, 2009.

[12] Kasik D J, Buxton W, Ferguson D R. Ten CAD challenges [J]. IEEE Computer Graphics and Applications, 2005, 25 (2): 81–92.

[13] Brunnermeier S B, Martin S A. Interoperability costs in the US automotive supply chain [J]. Supply Chain Management: An International Journal, 2002, 7 (2): 71–82.

[14] Hughes T J R, Cottrell J A, Bazilevs Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement [J]. Computer Methods in Applied Mechanics and Engineering, 2005, 194 (39–41): 4135–4195.

[15] Kiendl J, Bletzinger K-U, Linhard J, et al. Isogeometric shell analysis with Kirchhoff–Love elements [J]. Computer Methods in Applied Mechanics and Engineering, 2009, 198 (49-52): 3902–3914.

[16] Benson D, Bazilevs Y, Hsu M-C, et al. Isogeometric shell analysis: the Reissner–Mindlin shell [J].

Computer Methods in Applied Mechanics and Engineering, 2010, 199 (5-8): 276–289.

[17] Bazilevs Y, Calo V, Cottrell J, et al. Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows [J]. Computer Methods in Applied Mechanics and Engineering, 2007, 197 (1–4): 173–201.

[18] Akkerman I, Bazilevs Y, Calo V, et al. The role of continuity in residual-based variational multiscale modeling of turbulence [J]. Computational Mechanics, 2008, 41: 371–378.

[19] Gomez H, Hughes T J R, Nogueira X, et al. Isogeometric analysis of the isothermal Navier–Stokes–Korteweg equations [J]. Computer Methods in Applied Mechanics and Engineering, 2010, 199 (25–28): 1828–1840.

[20] Bazilevs Y, Calo V M, Hughes T J R, et al. Isogeometric fluid-structure interaction: Theory, algorithms, and computations [J]. Computational Mechanics, 2008, 43 (1): 3–37.

[21] Bazilevs Y, Hsu M-C, Scott M-A. Isogeometric fluid–structure interaction analysis with emphasis on non-matching discretizations, and with application to wind turbines [J]. Computer Methods in Applied Mechanics and Engineering, 2012, 249: 28–41.

[22] Vázquez R, Buffa A. Isogeometric analysis for electromagnetic problems [J]. IEEE Transactions on Magnetics, 2010, 46 (8): 3305–3308.

[23] Dölz J, Harbrecht H, Kurz S, et al. Bembel: The fast isogeometric boundary element C++ library for Laplace, Helmholtz, and electric wave equation [J]. SoftwareX, 2020, 11: 100476.

[24] Bazilevs Y, Calo V M, Zhang Y, et al. Isogeometric fluid–structure interaction analysis with applications to arterial blood flow [J]. Computational Mechanics, 2006, 38: 310–322.

[25] Morganti S, Auricchio F, Benson D, et al. Patient-specific isogeometric structural analysis of aortic valve closure [J]. Computer Methods in Applied Mechanics and Engineering, 2015, 284: 508–520.

[26] Qian K, Pawar A, Liao A, et al. Modeling neuron growth using isogeometric collocation based phase field method [J]. Scientific Reports, 2022, 12 (1): 8120.

[27] Wall W A, Frenzel M A, Cyron C. Isogeometric structural shape optimization [J]. Computer Methods in Applied Mechanics and Engineering, 2008, 197 (33–40): 2976–2988.

[28] Qian X. Full analytical sensitivities in NURBS based isogeometric shape optimization [J]. Computer Methods in Applied Mechanics and Engineering, 2010, 199 (29–32): 2059–2071.

[29] Seo Y-D, Kim H-J, Youn S-K. Isogeometric topology optimization using trimmed spline surfaces [J]. Computer Methods in Applied Mechanics and Engineering, 2010, 199 (49-52): 3270–3296.

[30] Hou W, Gai Y, Zhu X, et al. Explicit isogeometric topology optimization using moving morphable components [J]. Computer Methods in Applied Mechanics and Engineering, 2017, 326: 694–712.

[31] Wang Y, Wang Z, Xia Z, et al. Structural design optimization using isogeometric analysis: A comprehensive review [J]. Computer Modeling in Engineering & Sciences, 2018, 117 (3): 455–507.

[32] Gao J, Xiao M, Zhang Y, et al. A comprehensive review of isogeometric topology optimization:

Methods, applications and prospects [J]. Chinese Journal of Mechanical Engineering, 2020, 33 (1): 1–14.

[33] Gao J, Gao L, Xiao M. Isogeometric Topology Optimization: Methods, Applications and Implementations [M]. Springer Nature, 2022.

[34] Sande E, Manni C, Speleers H. Explicit error estimates for spline approximation of arbitrary smoothness in isogeometric analysis [J]. Numerische Mathematik, 2020, 144 (4): 889–929.

[35] De Lorenzis L, Wriggers P, Hughes T J R. Isogeometric contact: A review [J]. GAMM-Mitteilungen, 2014, 37 (1): 85–123.

[36] Yan J, Lin S, Bazilevs Y, et al. Isogeometric analysis of multi-phase flows with surface tension and with application to dynamics of rising bubbles [J]. Computers & Fluids, 2019, 179: 777–789.

[37] Kapl M, Vitrih V, Jüttler B, et al. Isogeometric analysis with geometrically continuous functions on two-patch geometries [J]. Computers & Mathematics with Applications, 2015, 70 (7): 1518–1538.

[38] Sogn J, Takacs S. Robust multigrid solvers for the biharmonic problem in isogeometric analysis [J]. Computers & Mathematics with Applications, 2019, 77 (1): 105–124.

[39] Dornisch W, Klinkel S, Simeon B. Isogeometric Reissner–Mindlin shell analysis with exactly calculated director vectors [J]. Computer Methods in Applied Mechanics and Engineering, 2013, 253: 491–504.

[40] Vuong A-V, Giannelli C, Jüttler B, et al. A hierarchical approach to adaptive local refinement in isogeometric analysis [J]. Computer Methods in Applied Mechanics and Engineering, 2011, 200 (49-52): 3554–3567.

[41] Giannelli C, Jüttler B, Kleiss S K, et al. THB-splines: An effective mathematical technology for adaptive refinement in geometric design and isogeometric analysis [J]. Computer Methods in Applied Mechanics and Engineering, 2016, 299: 337–365.

[42] Bazilevs Y, Calo V M, Cottrell J A, et al. Isogeometric analysis using T-splines [J]. Computer Methods in Applied Mechanics and Engineering, 2010, 199 (5–8): 229–263.

[43] Scott M A, Li X, Sederberg T W, et al. Local refinement of analysis-suitable T-splines [J]. Computer Methods in Applied Mechanics and Engineering, 2012, 213–216: 206–222.

[44] Li X, Scott M A. Analysis-suitable T-splines: Characterization, refineability, and approximation [J]. Mathematical Models and Methods in Applied Sciences, 2014, 24 (06): 1141–1164.

[45] 徐岗, 李新, 黄章进, et al. 面向等几何分析的几何计算 [J]. 计算机辅助设计与图形学学报, 2015, 27 (4): 570–581.

[46] Otoguro Y, Takizawa K, Tezduyar T E, et al. Turbocharger turbine and exhaust manifold flow computation with the space–time variational multiscale method and isogeometric analysis [J]. Computers & Fluids, 2019, 179: 764–776.

[47] Bazilevs Y, Akkerman I. Large eddy simulation of turbulent Taylor–Couette flow using isogeometric analysis and the residual-based variational multiscale method [J]. Journal of Computational

Physics, 2010, 229 (9): 3402–3414.

[48] Cohen E, Martin T, Kirby R, et al. Analysis-aware modeling: Understanding quality considerations in modeling for isogeometric analysis [J]. Computer Methods in Applied Mechanics and Engineering, 2010, 199 (5–8): 334–356.

[49] Xu G, Mourrain B, Duvigneau R, et al. Optimal analysis-aware parameterization of computational domain in 3D isogeometric analysis [J]. Computer-Aided Design, 2013, 45 (4): 812–821.

[50] Pilgerstorfer E, Jüttler B. Bounding the influence of domain parameterization and knot spacing on numerical stability in Isogeometric Analysis [J]. Computer Methods in Applied Mechanics and Engineering, 2014, 268: 589–613.

[51] Farin G, Hansford D. Discrete Coons patches [J]. Computer Aided Geometric Design, 1999, 16 (7): 691–700.

[52] Gravesen J, Evgrafov A, Nguyen D-M, et al. Planar parametrization in isogeometric analysis [C]. In Mathematical Methods for Curves and Surfaces: 8th International Conference, MMCS 2012, Oslo, Norway, June 28–July 3, 2012, Revised Selected Papers 8, 2014: 189–212.

[53] Xu G, Mourrain B, Duvigneau R, et al. Parameterization of computational domain in isogeometric analysis: Methods and comparison [J]. Computer Methods in Applied Mechanics and Engineering, 2011, 200 (23–24): 2021–2031.

[54] Wang X, Qian X. An optimization approach for constructing trivariate B-spline solids [J]. Computer-Aided Design, 2014, 46: 179–191.

[55] Xu G, Kwok T-H, Wang C C. Isogeometric computation reuse method for complex objects with topology-consistent volumetric parameterization [J]. Computer-Aided Design, 2017, 91: 1–13.

[56] Ugalde I A, Mederos V H, Sánchez P B, et al. Injectivity of B-spline biquadratic maps [J]. Computer Methods in Applied Mechanics and Engineering, 2018, 341: 586–608.

[57] Xu G, Mourrain B, Duvigneau R, et al. Constructing analysis-suitable parameterization of computational domain from CAD boundary by variational harmonic method [J]. Journal of Computational Physics, 2013, 252: 275–289.

[58] Nguyen T, Jüttler B. Parameterization of contractible domains using sequences of harmonic maps [C]. In International Conference on Curves and Surfaces, 2010: 501–514.

[59] Falini A, Špeh J, Jüttler B. Planar domain parameterization with THB-splines [J]. Computer Aided Geometric Design, 2015, 35: 95–108.

[60] Pan M-D, Chen F-L. Low-rank parameterization of volumetric domains for isogeometric analysis [J]. Computer-Aided Design, 2019, 114: 82–90.

[61] 张珊珊, 潘茂东, 陈发来. 基于拟共形映射的三角 Bézier 参数化 [J]. 计算机辅助设计与图形学学报.

[62] Hormann K, Greiner G. MIPS: An efficient global parametrization method [J]. Curve and Surface Design: Saint-Malo 1999, 2000: 153–162.

[63] Fu X-M, Liu Y, Guo B-N. Computing locally injective mappings by advanced MIPS [J]. ACM Transactions on Graphics (TOG), 2015, 34 (4): 1–12.

[64] Su J-P, Fu X-M, Liu L-G. Practical foldover-free volumetric mapping construction [C]. In Computer Graphics Forum, 2019: 287–297.

[65] Zheng Y, Chen F-L. Volumetric parameterization with truncated hierarchical B-splines for isogeometric analysis [J]. Computer Methods in Applied Mechanics and Engineering, 2022, 401: 115662.

[66] Garanzha V, Kaporin I. Regularization of the barrier variational method [J]. Computational Mathematics and Mathematical Physics, 1999, 39 (9): 1426–1440.

[67] Garanzha V, Kaporin I, Kudryavtseva L, et al. Foldover-free maps in 50 lines of code [J]. ACM Transactions on Graphics (TOG), 2021, 40 (4): 1–16.

[68] Wang X, Ma W. Smooth analysis-suitable parameterization based on a weighted and modified Liao functional [J]. Computer-Aided Design, 2021, 140: 103079.

[69] Martin T, Cohen E, Kirby R M. Volumetric parameterization and trivariate B-spline fitting using harmonic functions [J]. Computer Aided Geometric Design, 2009, 26 (6): 648–664.

[70] Shamanskiy A, Gfrerer M H, Hinz J, et al. Isogeometric parametrization inspired by large elastic deformation [J]. Computer Methods in Applied Mechanics and Engineering, 2020, 363: 112920.

[71] Ali Z, Ma W. Isogeometric collocation method with intuitive derivative constraints for PDE-based analysis-suitable parameterizations [J]. Computer Aided Geometric Design, 2021, 87: 101994.

[72] Hinz J, Möller M, Vuik C. Elliptic grid generation techniques in the framework of isogeometric analysis applications [J]. Computer Aided Geometric Design, 2018, 65: 48–75.

[73] Hinz J, Möller M, Vuik C. Spline-based parameterization techniques for twin-screw machine geometries [C]. In IOP Conference Series: Materials Science and Engineering, 2018: 012030.

[74] Hinz J P. PDE-Based Parameterization Techniques for Isogeometric Analysis Applications [D]. [S. l.]: Delft University of Technology, 2020.

[75] Xu J-L, Chen F-L, Deng J-S. Two-dimensional domain decomposition based on skeleton computation for parameterization and isogeometric analysis [J]. Computer Methods in Applied Mechanics and Engineering, 2015, 284: 541–555.

[76] Bastl B, Slabá K. Planar multi-patch domain parameterization for isogeometric analysis based on evolution of fat skeleton [J]. Computer Methods in Applied Mechanics and Engineering, 2021, 385: 114045.

[77] Xu G, Li M, Mourrain B, et al. Constructing IGA-suitable planar parameterization from complex CAD boundary by domain partition and global/local optimization [J]. Computer Methods in Applied Mechanics and Engineering, 2018, 328: 175–200.

[78] Xiao S-W, Kang H-M, Fu X-M, et al. Computing IGA-suitable planar parameterizations by PolySquare-enhanced domain partition [J]. Computer Aided Geometric Design, 2018, 62: 29–43.

[79] Wang S-Y, Ren J, Fang X, et al. IGA-suitable planar parameterization with patch structure simpli-

fication of closed-form polysquare [J]. Computer Methods in Applied Mechanics and Engineering, 2022, 392: 114678.

[80] Xu G, Mourrain B, Duvigneau R, et al. Analysis-suitable volume parameterization of multi-block computational domain in isogeometric applications [J]. Computer-Aided Design, 2013, 45 (2): 395–404.

[81] Lin H, Huang H, Hu C. Trivariate B-spline solid construction by pillow operation and geometric iterative fitting [J]. Science China Information Sciences, 2018, 61 (11): 118103.

[82] Haberleitner M, Jüttler B, Masson Y. Isogeometric segmentation via midpoint subdivision suitable solids [J]. Computer-Aided Design, 2019, 114: 179–190.

[83] Chen L, Xu G, Wang S-Y, et al. Constructing volumetric parameterization based on directed graph simplification of l_1 polycube structure from complex shapes [J]. Computer Methods in Applied Mechanics and Engineering, 2019, 351: 422–440.

[84] Chen L, Bu N-Y, Jin Y, et al. Construction of IGA-suitable volume parametric models by the Segmentation–Mapping–Merging mechanism of design features [J]. Computer-Aided Design, 2022, 146: 103228.

[85] Zhang Y, Wang W, Hughes T J R. Solid T-spline construction from boundary representations for genus-zero geometry [J]. Computer Methods in Applied Mechanics and Engineering, 2012, 249–252: 185–197.

[86] Zhang Y, Wang W, Hughes T J R. Conformal solid T-spline construction from boundary T-spline representations [J]. Computational Mechanics, 2013, 51 (6): 1051–1059.

[87] Pan M-D, Chen F-L. Constructing planar domain parameterization with HB-splines via quasi-conformal mapping [J]. Computer Aided Geometric Design, 2022, 97: 102133.

[88] Chan C L, Anitescu C, Rabczuk T. Volumetric parametrization from a level set boundary representation with PHT-splines [J]. Computer-Aided Design, 2017, 82: 29–41.

[89] Zhu X, Ji Y, Zhu C-G, et al. Isogeometric analysis for trimmed CAD surfaces using multi-sided toric surface patches [J]. Computer Aided Geometric Design, 2020, 79: 101847.

[90] Li J-G, Ji Y, Zhu C-G. De Casteljau algorithm and degree elevation of toric surface patches [J]. Journal of Systems Science and Complexity, 2021, 34 (1): 21–46.

[91] Ji Y, Li J-G, Yu Y-Y, et al. h-Refinement method for toric parameterization of planar multi-sided computational domain in isogeometric analysis [J]. Computer Aided Geometric Design, 2022, 93: 102065.

[92] Wang Z-H, Cao J, Wei X-D, et al. TCB-spline-based isogeometric analysis method with high-quality parameterizations [J]. Computer Methods in Applied Mechanics and Engineering, 2022, 393: 114771.

[93] Wang Z, Cao J, Wei X, et al. Kirchhoff–Love shell representation and analysis using triangle configuration B-splines [J]. Computer Methods in Applied Mechanics and Engineering, 2023, 416:

116316.

[94] Pan Q, Rabczuk T, Yang X. Subdivision-based isogeometric analysis for second order partial differential equations on surfaces [J]. Computational Mechanics, 2021, 68 (5): 1205–1221.

[95] Xie J, Xu J-L, Dong Z, et al. Interpolatory Catmull-Clark volumetric subdivision over unstructured hexahedral meshes for modeling and simulation applications [J]. Computer Aided Geometric Design, 2020, 80: 101867.

[96] Forsey D R, Bartels R H. Hierarchical B-spline refinement [C]. In Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques, 1988: 205–212.

[97] Giannelli C, Jüttler B, Speleers H. THB-splines: The truncated basis for hierarchical splines [J]. Computer Aided Geometric Design, 2012, 29 (7): 485–498.

[98] Carraturo M, Giannelli C, Reali A, et al. Suitably graded THB-spline refinement and coarsening: Towards an adaptive isogeometric analysis of additive manufacturing processes [J]. Computer Methods in Applied Mechanics and Engineering, 2019, 348: 660–679.

[99] Wei X-D, Zhang Y, Liu L, et al. Truncated T-splines: Fundamentals and methods [J]. Computer Methods in Applied Mechanics and Engineering, 2017, 316: 349–372.

[100] Liu L, Casquero H, Gomez H, et al. Hybrid-degree weighted T-splines and their application in isogeometric analysis [J]. Computers & Fluids, 2016, 141: 42–53.

[101] Huang W, Russell R D. Moving mesh strategy based on a gradient flow equation for two-dimensional problems [J]. SIAM Journal on Scientific Computing, 1998, 20 (3): 998–1015.

[102] Huang W, Russell R D. Adaptive moving mesh methods [M]. Springer Science & Business Media, 2010.

[103] 徐岗, 朱亚光, 邓立山, et al. 局部误差驱动的等几何分析计算域自适应优化方法 [J]. 计算机辅助设计与图形学学报, 2014, 26 (10): 1633–1638.

[104] Xu G, Li B, Shu L, et al. Efficient r-adaptive isogeometric analysis with Winslow's mapping and monitor function approach [J]. Journal of Computational and Applied Mathematics, 2019, 351: 186–197.

[105] Ali Z, Ma W. PDE-Based and Solution-Dependent Parameterization for Isogeometric Analysis [C]. In 14th World Congress on Computational Mechanics (WCCM XIV), 8th European Congress on Computational Methods in Applied Science and Engineering (ECCOMAS 2020), 2021.

[106] Ji Y, Wang M-Y, Yu Y-Y, et al. Curvature-based r-adaptive isogeometric analysis with injectivity-preserving multi-sided domain parameterization [J]. Journal of Systems Science and Complexity, 2023, 36 (1): 53–76.

[107] Manni C, Speleers H. Geometric Challenges in Isogeometric Analysis [M]. Springer Nature, 2022.

[108] Buffa A, Gantner G, Giannelli C, et al. Mathematical foundations of adaptive isogeometric analysis [J]. Archives of Computational Methods in Engineering, 2022, 29 (7): 4479–4555.

[109] Dörfel M R, Jüttler B, Simeon B. Adaptive isogeometric analysis by local h-refinement with T-

splines [J]. Computer Methods in Applied Mechanics and Engineering, 2010, 199 (5-8): 264–275.

[110] Wang P, Xu J-L, Deng J-S, et al. Adaptive isogeometric analysis using rational PHT-splines [J]. Computer-Aided Design, 2011, 43 (11): 1438–1448.

[111] Kovacevic A, Rane S. Challenges in 3D CFD Modelling of Rotary Positive Displacement Machines [C]. In Journal of Physics: Conference Series, 2021: 012063.

[112] Kestelman H. Mappings with non-vanishing Jacobian [J]. The American Mathematical Monthly, 1971, 78 (6): 662–663.

[113] Zheng Y, Chen F-L. Volumetric boundary correspondence for Isogeometric Analysis based on unbalanced optimal transport [J]. Computer-Aided Design, 2021, 140: 103078.

[114] Nian X-S, Chen F-L. Planar domain parameterization for isogeometric analysis based on Teichmüller mapping [J]. Computer Methods in Applied Mechanics and Engineering, 2016, 311: 41–55.

[115] Mørken K. Some identities for products and degree raising of splines [J]. Constructive Approximation, 1991, 7 (1): 195–208.

[116] Xu G, Mourrain B, Duvigneau R, et al. Optimal analysis-aware parameterization of computational domain in isogeometric analysis [C]. In International Conference on Geometric Modeling and Processing, 2010: 236–254.

[117] Lyche T, Morken K. Spline methods draft [R]. 2008.

[118] Rado T. Aufgabe 41 [J]. Jahresber. Deutsch. Math.-Verein, 1926, 35: 49.

[119] Kneser H. Losung der aufgabe 41 [J]. Jahresber. Deutsche Meth., 1926: 123–124.

[120] Duren P, Hengartner W. Harmonic mappings of multiply connected domains [J]. Pacific Journal of Mathematics, 1997, 180 (2): 201–220.

[121] do Carmo M. Differential Geometry of Curves and Surfaces [M]. Prentice-Hall, Englewood Cliffs, New Jersey, 1976.

[122] Johnson S G. The NLopt nonlinear-optimization package. 2020.

[123] Nocedal J. Updating quasi-Newton matrices with limited storage [J]. Mathematics of Computation, 1980, 35 (151): 773–782.

[124] Liu D C, Nocedal J. On the limited memory BFGS method for large scale optimization [J]. Mathematical Programming, 1989, 45 (1): 503–528.

[125] Lin H, Jin S, Hu Q, et al. Constructing B-spline solids from tetrahedral meshes for isogeometric analysis [J]. Computer Aided Geometric Design, 2015, 35–36: 109–120.

[126] Yuan G-J, Liu H, Su J-P, et al. Computing planar and volumetric B-spline parameterizations for IGA by robust mapping fitting [J]. Computer Aided Geometric Design, 2021, 86: 101968.

[127] Ji Y, Yu Y-Y, Wang M-Y, et al. Constructing high-quality planar NURBS parameterization for isogeometric analysis by adjustment control points and weights [J]. Journal of Computational and Applied Mathematics, 2021, 396: 113615.

[128] Moré J J, Thuente D J. Line search algorithms with guaranteed sufficient decrease [J]. ACM Transactions on Mathematical Software (TOMS), 1994, 20 (3): 286–307.

[129] Arne V. Interface to a LBFGS solver. MATLAB Central File Exchange. 2021.

[130] Byrd R H, Nocedal J, Waltz R A. Knitro: An integrated package for nonlinear optimization [J]. Large-scale nonlinear optimization, 2006: 35–59.

[131] Ji Y, Wang M-Y, Wang Y, et al. Curvature-based r-adaptive planar NURBS parameterization method for isogeometric analysis using bi-level approach [J]. Computer-Aided Design, 2022, 150: 103305.

[132] Ji Y, Wang M-Y, Pan M-D, et al. Penalty function-based volumetric parameterization method for isogeometric analysis [J]. Computer Aided Geometric Design, 2022, 94: 102081.

[133] Anderson D G. Iterative procedures for nonlinear integral equations [J]. Journal of the ACM (JACM), 1965, 12 (4): 547–560.

[134] Anderson D G. Comments on "Anderson acceleration, mixing and extrapolation" [J]. Numerical Algorithms, 2019, 80 (1): 135–234.

[135] Toth A, Kelley C T. Convergence analysis for Anderson acceleration [J]. SIAM Journal on Numerical Analysis, 2015, 53 (2): 805–819.

[136] Evans C, Pollock S, Rebholz L G, et al. A proof that Anderson acceleration improves the convergence rate in linearly converging fixed-point methods (but not in those converging quadratically) [J]. SIAM Journal on Numerical Analysis, 2020, 58 (1): 788–810.

[137] Pollock S, Rebholz L G, Xiao M. Anderson-accelerated convergence of Picard iterations for incompressible Navier–Stokes equations [J]. SIAM Journal on Numerical Analysis, 2019, 57 (2): 615–637.

[138] Peng Y, Deng B, Zhang J, et al. Anderson acceleration for geometry optimization and physics simulation [J]. ACM Transactions on Graphics (TOG), 2018, 37 (4): 1–14.

[139] Zhang J, Peng Y, Ouyang W, et al. Accelerating ADMM for efficient simulation and optimization [J]. ACM Transactions on Graphics (TOG), 2019, 38 (6): 1–21.

[140] Hinz J, Buffa A. PDE-based parameterisation techniques for planar multipatch domains [J]. arXiv preprint, 2023: arXiv:2307.06265.

[141] Pratapa P P, Suryanarayana P, Pask J E. Anderson acceleration of the Jacobi iterative method: An efficient alternative to Krylov methods for large, sparse linear systems [J]. Journal of Computational Physics, 2016, 306: 43–54.

[142] Walker H F, Ni P. Anderson acceleration for fixed-point iterations [J]. SIAM Journal on Numerical Analysis, 2011, 49 (4): 1715–1735.

[143] Bian W, Chen X, Kelley C. Anderson acceleration for a class of nonsmooth fixed-point problems [J]. SIAM Journal on Scientific Computing, 2021, 43 (5): S1–S20.

[144] Chen K-W, Vuik C. Composite Anderson acceleration method with two window sizes and optimized damping [J]. International Journal for Numerical Methods in Engineering, 2022, 123 (23): 5964–

5985.

[145] Jüttler B, Langer U, Mantzaflaris A, et al. Geometry + simulation modules: Implementing isogeometric analysis [J]. PAMM, 2014, 14 (1): 961–962.

[146] Mantzaflaris A. An overview of geometry plus simulation modules [C]. In International Conference on Mathematical Aspects of Computer and Information Sciences, 2019: 453–456.

[147] Guennebaud G, Jacob B, et al. Eigen v3.4.0. 2021.

[148] Chen K-W, Vuik C. Non-stationary Anderson acceleration with optimized damping [J]. arXiv preprint, 2022: arXiv:2202.05295.

[149] Stosic N, Smith I, Kovacevic A. Screw Compressors: Mathematical Modelling and Performance Calculation [M]. Springer Science & Business Media, 2005.

[150] Kovacevic A, Stosic N, Smith I. Screw Compressors: Three Dimensional Computational Fluid Dynamics and Solid Fluid Interaction [M]. Springer Science & Business Media, 2007.

[151] TwinMesh. Reliable CFD analysis of rotary positive displacement machines. 2023.

[152] SCORG. Screw compressor rotor grid generator. 2023.

[153] ANSYS. ANSYS CFX: Turbomachinery CFD Software. 2023.

[154] Christensen P W, Klarbring A. An Introduction to Structural Optimization [M]. Springer Science & Business Media, 2008.

[155] Nocedal J, Wright S. Numerical Optimization [M]. Springer Science & Business Media, 2006.

[156] Gao X-F, Deng Z-F, Chen G-N. Hexahedral mesh re-parameterization from aligned base-complex [J]. ACM Transactions on Graphics (TOG), 2015, 34 (4): 1–10.

[157] Falini A, Jüttler B. THB–splines multi-patch parameterization for multiply-connected planar domains via template segmentation [J]. Journal of Computational and Applied Mathematics, 2019, 349: 390–402.

[158] Pan M, Zou R, Tong W, et al. G1-smooth planar parameterization of complex domains for isogeometric analysis [J]. Computer Methods in Applied Mechanics and Engineering, 2023, 417: 116330.

攻读博士学位期间科研项目及科研成果

已发表论文

[1] **Ye Ji**, Ying-Ying Yu, Meng-Yun Wang, & Chun-Gang Zhu. (2021). Constructing high-quality planar NURBS parameterization for isogeometric analysis by adjustment control points and weights. *Journal of Computational and Applied Mathematics*, 396, 113615. (中科院二区 Top 期刊, 本论文第3章)

[2] **Ye Ji**, Meng-Yun Wang, Mao-Dong Pan, Yi Zhang, & Chun-Gang Zhu. (2022). Penalty function-based volumetric parameterization method for isogeometric analysis. *Computer Aided Geometric Design*, 94, 102081. (中国数学会 T1 期刊, 本论文第4章)

[3] **Ye Ji**, Kewang Chen, Matthias Möller, & Cornelis Vuik. (2023). On an improved PDE-based elliptic parameterization method for isogeometric analysis using preconditioned Anderson acceleration. *Computer Aided Geometric Design*, 102, 102065. (中国数学会 T1 期刊, 本论文第5章)

[4] **Ye Ji**, Matthias Möller, & Hugo M. Verhelst. Design through analysis. *Book Chapter in Fluids under Control*. Birkhäuser Publishing Ltd. 2023. Accepted. (`https://link-springer-com.tudelft.idm.oclc.org/book/9783031473548`) (预计于 2024 年初出版, Hardcover ISBN: 978-3-031-47354-8, 本论文第5章)

[5] **Ye Ji** & Matthias Möller. Mesh generation for twin-screw compressors by spline-based parameterization using preconditioned Anderson acceleration. *In: 13th International Conference on Compressors and their Systems.* 2023. Accepted. (`https://link.springer.com/book/9783031426629`) (预计于 2024 年初出版, Hardcover ISBN: 978-3-031-42662-9, 本论文第6章)

[6] **Ye Ji**, Meng-Yun Wang, Yu Wang, & Chun-Gang Zhu. (2022). Curvature-based r-adaptive planar NURBS parameterization method for isogeometric analysis using bi-level approach. *Computer-Aided Design*, 150, 103305. (CCF-B 类期刊, 本论文第7章)

[7] **Ye Ji**, Jing-Gai Li, Ying-Ying Yu, & Chun-Gang Zhu. (2022). h-Refinement method for toric parameterization of planar multi-sided computational domain in isogeometric analysis. *Computer Aided Geometric Design*, 93, 102065. (中国数学会 T1 期刊)

[8] **Ye Ji**, Meng-Yun Wang, Ying-Ying Yu, & Chun-Gang Zhu. (2022). Curvature-based r-adaptive isogeometric analysis with injectivity-preserving multi-sided domain parameterization. *Journal of Systems Science and Complexity*, 36(1), 53–76. (中国数学会 T2 期刊)

[9] Ying-Ying Yu, **Ye Ji**, & Chun-Gang Zhu. (2023). Sufficient condition for injectivity of NURBS volumes by tangent cones. *Journal of Computational and Applied Mathematics*, 432, 115303. (中科院二区 Top 期刊)

[10] Meng-Yun Wang, **Ye Ji**, & Chun-Gang Zhu. Degree elevation and knot insertion for generalized Bézier surfaces and their application to isogeometric analysis. *Journal of Computational Mathematics*, accepted. (中国数学会 T1 期刊)

[11] Ying-Ying Yu, **Ye Ji**, Jing-Gai Li, & Chun-Gang Zhu. (2021). Conditions for injectivity of toric volumes with arbitrary positive weights. *Computers & Graphics*, 97, 88–98. (CCF-C 类期刊)

[12] Jing-Gai Li, **Ye Ji**, & Chun-Gang Zhu. (2021). De Casteljau algorithm and degree elevation of toric surface patches. *Journal of Systems Science and Complexity*, 34(1), 21–46. (中国数学会 T2 期刊)

[13] Xue-Feng Zhu, **Ye Ji**, Chun-Gang Zhu, Ping Hu, & Zheng-Dong Ma. (2020). Isogeometric analysis for trimmed CAD surfaces using multi-sided toric surface patches. *Computer Aided Geometric Design*, 79, 101847. (中国数学会 T1 期刊)

[14] Ying-Ying Yu, **Ye Ji**, & Chun-Gang Zhu. (2020). An improved algorithm for checking the injectivity of 2D toric surface patches. *Computers & Mathematics with Applications*, 79(10), 2973–2986. (中科院二区 Top 期刊)

[15] Yu Wang, Lingzhi Jin, Hang Yang, Peng Hao, **Ye Ji**, & Bo Wang. (2023). Isogeometric-based mapping modeling and buckling analysis of stiffened panels. *Thin-Walled Structures*, 186, 110676. (中科院二区期刊)

在国际和国内学术会议上口头报告论文 (部分)

[1] Analysis-suitable parameterization techniques for isogeometric analysis. *RICAM Workshop on Topology Optimization and Isogeometric Analysis*. Linz, Austria. Sep. 11 – 13, 2023.

[2] Mesh generation for twin-screw compressors by spline-based parameterization using preconditioned Anderson acceleration. *The 13th International Conference on Compressors and their Systems*. London, the United Kingdom. Sep. 11 – 13, 2023.

[3] Fast and robust solvers for local/global domain parameterizations within G+Smo. *The 11th International Conference on IsoGeometric Analysis (IGA) 2023*. Lyon, France. Jun. 18 – 21, 2023.

[4] On an improved PDE-based elliptic parameterization method for isogeometric analysis using preconditioned Anderson acceleration. *International Conference on Geometric Modeling and Processing (GMP) 2023*. Genova, Italy. Jul. 6 – 7, 2023.

[5] Implementation of analysis-suitable parameterization construction using G+Smo. *G+Smo Developers' Days and preCICE meeting 2022*. Delft, the Netherlands. Oct. 10 – 14, 2022.

[6] Curvature-based r-adaptive planar NURBS parameterization method for isogeometric analysis using bi-level approach. *Symposium on Solid and Physical Modeling (SPM) 2022*. Online. Jun. 27 – 29, 2022.

[7] Penalty function-based volumetric parameterization method for isogeometric analysis. *International Conference on Geometric Modeling and Processing (GMP) 2022*. Okinawa, Japan (Online). May 11–13, 2022.

[8] High-quality planar NURBS parameterization based on alternating control points and weights optimization. 第十三届全国几何设计与计算学术会议 *(GDC)*. 长沙, 中国, 10 月 9 日 – 11 日, 2021.

[9] Curvature-based r-adaptive isogeometric analysis with injectivity-preserving multi-sided domain parameterization. 第十二届中国数学会计算机数学大会 *(CM)*. 桂林, 中国, 6 月 4 日 – 7 日, 2021.

[10] High-quality planar NURBS parameterization of computational domain in IGA via control points and weights optimization. 全国网格生成及应用研讨会 *(MEGAS)*. 杭州, 中国, 5 月 28 日 – 30 日, 2021.

获得奖励

[1] **Conference Best Paper Award**, *International Conference on Geometric Modeling and Processing (GMP) 2023*. Genova, Italy. Jul. 6 – 7, 2023. 完成人排序: 1/4.

[2] **SIAM Hackathon 2023 Challenge Winner** (Amazon Web Services Challenge), *SIAM Conference on Computational Science and Engineering (CSE23)*. Amsterdam, the Netherlands. Feb. 26 – Mar. 3, 2023. 完成人排序: 1/6.

[3] **Conference Best Paper Award**, *The International Conference on Computer-Aided Design and Computer Graphics (CAD/Graphics)*. Xi'an, China. May 15 – 16, 2021. 完成人排序: 2/4.

主持和参与的科研项目

[1] 国家建设高水平大学公派研究生项目 (202106060082): 基于 PDE 的等几何参数化方法及其在双螺杆压缩机械中的应用. 2021.10–2023.10. 负责人: 纪野.

[2] 国家自然科学基金面上项目 (12071057): 参数曲面/体的理论及其在等几何分析中的应用研究. 2021.01–2024.12. 负责人: 朱春钢.

[3] 国家自然科学基金面上项目 (11671068): 参数曲线曲面的几何性质研究. 2017.01–2020.12. 负责人: 朱春钢.

致　谢

时光如白驹过隙, 自 2013 年秋考入大连理工大学已经十年有余, 我依然记得第一次以本科生身份踏入校园时无比激动的心情. 在这片沃土上, 我有幸结识了许多才华横溢的师长和同学. 他们的出现不仅让我的求学之路更加充实和精彩, 也让我在成长的过程中不断积累经验、丰富知识、开拓视野. 我在这里结交了许多志同道合的朋友, 我们分享喜怒哀乐, 一同度过了无数个难忘的时刻.

首先, 我要向我的硕士和博士生导师, 朱春钢教授, 表示最诚挚的感谢. 初识朱老师是在本科必修课《数值逼近与计算几何》课上, 我仍然记得当时朱老师深入浅出的讲解, 将复杂的数值逼近与样条理论生动形象地呈现给我们. 朱老师清晰的逻辑思维和儒雅的气质深深地吸引了我, 使我在保送研究生时毫不犹豫地选择了他作为我的导师. 事实证明, 我的选择是正确的. 在我整个研究生生涯中, 朱老师都无私地关心和指导我, 让我在学习和科研的道路上走得越来越稳. 每次讨论和修改论文, 朱老师都非常认真, 并且充分尊重我的想法, 为我在学术研究中提供了充分的自由发挥空间, 助我将那些天马行空的想法一一付诸实现. 他不仅尊重和鼓励我在学术上独立思考, 还在关键时刻给予我强大的精神支持. 在我感觉快要屈服于压力之时, 他总是伸出援手为我解围, 让我重拾信心. 他的支持和鼓励成为了我不断前行的动力, 也让我更加坚信自己的能力和潜力. 此外, 朱老师还非常关心我的生活. 在我面临人生重大决策或困境时, 朱老师总是充分尊重我的想法并给予我支持. 如师如父, 亦师亦友. 与朱老师一同工作和生活的时光是我人生中不可多得的宝贵财富. 我由衷地感激朱老师在我的学术和人生里程中所给予的宝贵启示和无价支持, 他的悉心教诲将永远镌刻在我的心灵深处.

我衷心感激在大连理工大学遇到的所有老师, 特别是来自数学科学学院的苏志勋老师、董波老师、李崇君老师、刘秀平老师、杜磊老师、程明松老师、徐敏老师、张仁权老师、张旭平老师、赵国辉老师、王磊老师、鲁大伟老师和张超老师, 以及汽车工程学院的祝雪峰老师和夏阳老师等. 他们不仅为我奠定了扎实的学术基础, 还传授了许多人生的宝贵智慧. 正是他们无私的教诲和引领, 激发了我对知识的渴望和持续追求, 以及在学术领域内不断探索和拓展视野的勇气.

我衷心感谢国家留学基金委员会 (The China Scholarship Council, CSC) 在我博士生涯后两年的慷慨资助, 使我有机会来到荷兰代尔夫特理工大学继续深造, 拓宽我的学术视野. 在此, 我要向我的国外导师 Matthias Möller 教授表示最真挚的感谢. 他无私地为我提供学术指导和鼓励, 使我在国际学术领域更加自信地迈进. 在 Möller 教授的悉心教导下, 我的学术研究和专业能力得到了显著的提升. 同时, 我要感谢代尔夫特应用数学学院系主任 Cornelis Vuik 教授对我参加学术会议的鼎力支持. 在 Vuik 教授的帮助下, 我得以在这个挑战与机遇并存的国际学术舞台上拓展视野, 学习世界各地的先进

理念和方法, 结识志同道合的朋友, 与领域内的杰出学者交流, 并从他们身上汲取丰富的知识和经验. 感谢法国国家信息与自动化研究所 (INRIA) 的 Angelos Mantzaflaris 教授对我学习 G+Smo 库上的指导和无私帮助. 此外, 我要感谢伦敦大学城市学院 (City, University of London) 的 Ahmed Kovacevic 教授和 Sham Rane 博士在双螺杆压缩机项目上的无私帮助. 与他们的月度会议让我深受启发, 使我能够更好地了解实际工业场景的需求和挑战, 丰富了我的知识储备. 我深知这段时间的经历对于我的成长和发展具有重要意义, 我为能在这样一个充满智慧和激情的国际环境中学习而感到自豪.

向 Hugo Verhelst 表示衷心的感谢, 与他的讨论不仅使我受益良多, 他的幽默风趣也给我带来了许多欢乐. 同时, 我要感谢在代尔夫特学习期间遇到的每一位朋友和同事, 包括李骏、单秀杰、郇越、Leila Hashemi、Deepesh Toshniwal 教授、Gijs Mart、Merel A. Schalkers 和 Giorgio Tosti Balducci. 在你们的陪伴下, 我们一起学习和玩耍, 度过了许多珍贵的时光. 感谢你们带给我的温暖, 让我在远离家乡的代尔夫特找到了归属感.

我要特别感谢那些与我合作过以及在我的学术研究中给予过帮助的同行和学者们. 他们包括中国科学技术大学的陈发来教授、傅孝明教授、李新教授、年先顺博士和郑野博士; 杭州电子科技大学的徐岗教授、邓重阳教授和张蒙同学; 南京航空航天大学的潘茂东教授; 南京信息工程大学的陈克旺博士; 大连理工大学软件学院的雷娜教授; 以及大连理工大学工业装备结构分析国家重点实验室的郝鹏教授和王禹博士生. 他们的专业知识和丰富经验对我的研究工作产生了深远的影响. 在与这些杰出同行和学者的交流与合作中, 我不仅学到了许多宝贵的知识和技能, 也结识了众多优秀的人才. 他们的支持和鼓励使我在学术道路上越走越远, 不断克服困难, 实现更高的目标. 在此, 我向他们表达我最诚挚的感谢, 期待未来有更多的机会与他们共同探讨学术与疑问.

在此, 我要向实验室的同门们表示衷心的感谢. 他们在学术和生活上给予了我无尽的帮助和支持. 向已经毕业的师兄师姐们李彩云、张胜刚、孙兰银、赵轩艺、王涵、李敬改、于滢滢等表示感激; 感谢同级的同学们, 韩志红和方岩; 也要感谢热情的师弟师妹们周沛、王梦云、兰林、张艺、李琳琳、杨晶晶、毛雪娇等. 在实验室大家共同的努力下, 我们取得了许多值得骄傲的研究成果. 我们共同学习、交流和探讨问题, 结下了深厚的友谊. 在这个过程中我也不断地成长和进步. 能够与这样一群富有才华, 热情洋溢且乐于助人的同门一起奋斗, 是我莫大的荣幸!

除此之外, 我还要感谢我的同学们和朋友们, 特别是田国强、端木维向、张怡、张景晟、吴过、李萌等. 我们一起度过了无数美好而难忘的时光, 共同学习、交流和探讨问题, 携手应对各种挑战和困难. 正是他们陪伴和支持, 让我的求学之路充满了欢声笑语, 使我在忙碌的学术生活中找到了心灵的寄托. 他们的陪伴让我在人生的道路上不再孤单, 与他们共同经历过的每一刻都成为了我宝贵的回忆. 在他们的支持与关爱下, 我勇敢地面对生活中的种种挫折, 不断坚定信心、勇敢前行.

我的家人始终是我人生道路上坚定的支柱, 尤其是我的父母. 他们一直在我背后默默地付出, 用无言的支持和无尽的爱填充我的人生. 正因为有了他们不断的鼓励和坚定的支持, 我才能顺利地完成这篇博士论文. 从我记事起，这种被爱和支持包围的感觉就从未改变, 让我深感自己是世界上最幸运的孩子. 他们的无私关爱和持续鼓励, 让我有了面对生活各种挑战的勇气和力量, 使我能够迎难而上, 克服人生道路上的种种挑战. 在此, 我想对我的父母表达最深刻的感激, 他们的牺牲和付出是我不断学习进步和个人成长的最大动力. 谢谢你们，我亲爱的爸妈!

在此, 我想特别向我的妻子, 李娜女士, 表示最深的感激. 在异国他乡, 能遇到你是我莫大的幸运. 你是我最坚实的支持, 是你的陪伴和鼓励让我能够不断前行, 勇敢面对各种困难和压力. 每当工作和研究遇到瓶颈, 是你给予我信心和力量, 让我能重新站起来继续战斗. 每一个难以入睡的深夜, 每一个周末和假期, 你都在默默地付出和支持, 让我能专心完成我的研究. 你不仅是我的妻子, 更是我最好的朋友和最信任的合作伙伴. 对我而言, 你不仅是我的家庭, 更是我力量和勇气的源泉. 永远爱你!

在我的求学生涯中, 还有许多其他的人和事值得我感激. 虽然篇幅有限, 无法一一道谢, 但这些美好的回忆将永远地烙印在我的心中. 再次感谢所有在这段时间里给予我关心、帮助和支持的人们. 在未来的日子里, 我将怀揣这份感恩之情继续前行, 继续在未来的生活和工作上努力拼搏, 为自己的理想和梦想而努力, 勇敢地迈向未来.

最后, 祝愿伟大祖国繁荣昌盛、国泰民安!

作者简介

姓名: 纪 野

性别: 男

出生年月: 1995 年 09 月

民族: 汉

籍贯: 黑龙江省

研究方向: 等几何分析、计算机辅助几何设计

简历:

2019.09–2023.12	大连理工大学	数学科学学院	计算数学	博士研究生
2021.10–2023.12	代尔夫特理工大学	应用数学学院	数值分析	联合培养
2017.09–2019.08	大连理工大学	数学科学学院	计算数学	硕士研究生
2013.09–2017.07	大连理工大学	数学科学学院	信息与计算科学	本科